ADVANCED PROCESS CONTROL IN MANUFACTURING PROCESS WITH HIGH DIMENSIONAL MEASUREMENTS

A Dissertation
Presented to
The Academic Faculty

by

Zhen Zhong

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology
August 2021

AUTOMATIC FEEDBACK CONTROL IN HIGH PRECISION MANUFACUTRING

Approved by:

Dr. Jianjun Shi, Advisor
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Kamran Paynabar, Advisor
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Jing Li
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Chun Zhang
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

Dr. Lance Waller
Rollins School of Public Health
Emory University

Date Approved: 04/20/2021

ACKNOWLEDGEMENT

Throughout the writing of this dissertation, I have received a great deal of support and assistance.

First and foremost, I would like to thank my advisors, Professor Jianjun (Jan) Shi and Professor Kamran Paynabar, for their continuous presence, dedicated supervision, and support throughout my Ph.D. studies. Their expertise and insightful feedbacks were invaluable in formulating research questions and methodologies, sharpened my thoughts, and brought my work to a higher level. Besides research, their patience, kindness, and enthusiasm have made a significant impact on my personality. To put it in a nutshell, without their support, I would have not been here, and I could not imagine having better advisors for my Ph.D. study.

I would also like to thank my dissertation committee members, Professor Chun Zhang, Professor Jing Li, and Professor Lance Waller for their encouragement and insightful comments. I would like to express my gratitude for their continuous support and for evaluating my dissertation.

I would like to thank my collaborators Professor Xiaoming Huo, Professor Li-Hsiang Lin, Dr. Jeffrey H. Hunt, Mr. Hyunsik Kim, Dr. Andi Wang, Dr. Xiaowei Yue, Dr. Juan Du, Dr. Mostafa Reisi Gahrooei, Dr. Xinran Shi, Dr. Feng Wang, Mr. Shancong Mou, Mr. Dhari Alenezi, and Mr. Michael Biehler for the wonderful collaboration and kind support. Their knowledge and understanding of research inspired me to come up with innovative ideas.

I would like to express my gratitude to my friends for the discussions and suggestions, and for all the fun we have had in the last four years. They include but are not limited to Dr. Andi Wang, Dr. Xiaowei Yue, Dr. Juan Du, Dr. Mostafa Reisi Gahrooei, Dr. Xinran Shi, Dr. Feng Wang, Mr. Shancong Mou, Mr. Dhari Alenezi, and Mr. Michael Biehler. I would like to express my gratitude to Ms. Liping Luo for her kindness and tremendous care. Last but not the least, I would like to thank my parents and for their spiritual support throughout my doctoral studies and life in general. This dissertation stands as a testament to love and encouragement.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Automatic feedback or feedforward control has been widely used in manufacturing systems to reduce process variability and ensure on-target product quality. In this thesis, the methodologies of automatic control are further investigated to address model uncertainties, high-dimensional sensing feedback control, and their applications in challenging engineering problems. Motivated by real needs from current industrial production systems, three control methods are studied in this thesis in Chapters 2, 3, and 4.

In Chapter 2, an adaptive cautious regularized run-to-run control scheme is developed for overlay control in photolithography processes. Photolithography is the bottleneck for quality improvement in semiconductor manufacturing. The decreasing critical dimensions of the semiconductor product require more effective run-to-run control technology. Currently, Exponential Weighted Moving Average (EWMA) control scheme is widely used in the overlay control of lithography processes. In this chapter, three shortcomings of the current EWMA run-to-run control scheme are investigated: (i) the existing EWMA control scheme has its weight parameter λ set as a fixed value, which does not perform well when the process changes; (ii) the existing EWMA control scheme does not consider the model and parameter uncertainties in practice; and (iii) the adjustable range of the control variables is not considered in the existing EWMA control scheme. To address these limitations, we propose a new adaptive, cautious, and optimal run-to-run control scheme. The effectiveness of the new controller is validated through surrogated simulation studies.

In Chapter 3, an image-based feedback control strategy is developed by using tensor representation and analysis. The problem is motivated by the photolithography process, where the system output is image signals measuring the overlay error, and the control inputs are tuning vectors. To develop a control strategy, one first needs to off-line estimate the process model by finding the relationship between the image output and vector inputs, and then to obtain the control law by online minimizing the control objective function. The main challenges in achieving such a control strategy include (i) the high dimensionality of the output in building a regression model, (ii) the spatial structure of image outputs and the temporal structure of the image sequence, and (iii) non-i.i.d noises. To address these challenges, we propose a novel tensor-based process control approach by incorporating the tensor time series and regression techniques. Based on the process model, we obtain the control law by minimizing a control objective function. Although our proposed approach is demonstrated with the 2D images as the system output, it can have the potential to be extended to the higher-order tensors such as video signals or point cloud data. Simulation and case studies show that our proposed method is more effective than benchmark methods in terms of relative mean square error.

Chapter 4 will investigate how to achieve half-fuselage assembly via active control. In a half fuselage assembly process, shape control is vital for achieving ultra-high precision assembly. To achieve better shape adjustment, we need to determine the optimal location and force of each actuator to push and pull a fuselage to compensate for its initial shape distortion. The current practice achieves this goal by solving a surrogate model-based optimization problem. However, there are two limitations in this surrogate model-based method: (1) Low efficiency: Collecting training data for surrogate modeling from many

FEA replications is time-consuming. (2) Non-optimality: The required number of FEA replications for building an accurate surrogate model will increase as the potential number of actuator locations increases. Therefore, the surrogate model can only be built on a limited number of prespecified potential actuator locations, which will lead to sub-optimal control results. To address these issues, this chapter proposes an FEA model-based automatic optimal shape control (AOSC) framework. This method directly loads the system equation from the FEA simulation platform to determine the optimal location and force of each actuator. Moreover, the proposed method further integrates the cautious control concept into the AOSC system to address model uncertainties in practice. The case study with industrial settings shows that the proposed Cautious AOSC method achieves higher control accuracy compared to current industrial practice.

CHAPTER 1. INTRODUCTION

In manufacturing systems, system control is widely adopted to ensure on-target quality and to reduce process variability. To achieve this, we need to adjust a set of control variables, which will significantly influence the response/quality measure. A satisfactory control strategy generates a set of control variables such that the deviation of a quality measure from its target is minimized. For example, in semiconductor manufacturing, we need multiple stages to transfer wafers into final products. However, among all of these stages, the bottleneck of the quality improvement is lithography, which carves pre-specified 2-D patterns on wafer surfaces through optical systems. Since the lithography is conducted layer by layer, there are some pattern misalignments between two adjacent layers. This misalignment is known as the overlay error. Overlay errors are usually induced by biased wafer position, inappropriate settings of optical systems, etc. This, however, can be corrected by changing the settings on the lithography machine. Our objective is to find a proper control strategy that can generate a set of machine settings based on the past overlay error images to minimize the next time overlay error. Other application examples in which automatic feedback control is important, include fuselage assembly (Yue et al. 2018), hot rolling (Yan et al. 2015), and additive manufacturing (Liu et al., 2019).

To design an effective automatic feedback control strategy, we first need to estimate the process model that describes the relationship between output and control variables offline, then obtain the control law by optimizing the control objective function online. In the literature, there are some researches on the automatic feedback control scheme for manufacturing output measurement data. Tseng et al., (2002) proposed the multivariate

controller for multiple-input, multiple-output (MIMO) manufacturing processes. Castillo et al. (2002) proposed a multivariate double EWMA controller for drifting processes. Moreover, Tseng et al. (2013) proposed a multivariate EWMA controller for a linear dynamic process. Liu et al., (2019) proposed an image-based control method for additive manufacturing. However, all these control algorithms have at least one of the following drawbacks:

1. All of these methods have some issues in obtaining an accurate process model. They either assume the process model is known or construct the model purely based on engineering features. However, in real applications, the process model is usually unknown. Moreover, constructing a model purely based on engineering features has drawbacks. First, to extract engineering features, we need to have sufficient domain knowledge, which may not be available in many applications. Second, constructing a model purely based on engineering features will lose some essential information contained in measurement data. This will lead to an inaccurate process model and consequently, to poor control performance.
2. Even with the accurate process model, all of the above-mentioned methods have some drawbacks in calculating an appropriate control law. There are mainly three limitations: (i) existing methods cannot adjust adaptively according to the process changes; (ii) these methods fail to take the model and parameter uncertainties into consideration; and (iii) the adjustable range of the control variables is not considered in these methods.

In this thesis, we conducted three studies to tackle these drawbacks. For the first study, we focus on semiconductor manufacturing applications. In particular, we propose

an adaptive, cautious, and regularized control scheme to improve the overall control performance. This method that can be adaptive to process changes, considers the uncertainties of the process gain and incorporates the constraints for the control parameters' range. The improvements are achieved by integrating three algorithms: (i) the adaptive algorithms, which can make self-adjustment as process changes by learning the tuning parameters based on historical data; (ii) cautious control algorithm, which takes the model uncertainty into consideration and generates control laws by minimizing the expectation of the control objective function; and (iii) regularized control algorithms, which generate the optimal control law when there are boundary constraints on control variables. This is achieved by solving an optimization problem consisting of the original control objective function as well as the boundary constraints. The effectiveness of our proposed method will be demonstrated in a series of simulation studies designed from real system setups.

In the second study, on the other hand, we try to propose an overarching methodology for designing and deploying an optimal control strategy that can handle High-Dimensional (HD) output and both Low-Dimensional (LD) and HD control variables. This is comprised of two steps. In the offline estimation step, there are mainly three challenges: 1) the high-dimensionality of the model coefficients and the response variable, which may lead to overfitting; 2) the Spatio-temporal structure of the response sequence, and 3) the non-iid noises in the system. We develop a novel tensor-based regression/time-series modeling to address these challenges. For the online control part, we use an optimization model with a squared loss to obtain the optimal control law. Simulations and case studies show that our proposed method is more effective than the existing methods.

In the third study, we propose a novel process model by considering engineering domain knowledge in the half-fuselage assembly process. Compared to traditional surrogate model-based methods, our proposed method can formulate the problem based on exact engineering physical constraints, which leads to less computational effort and higher accuracy. Based on the novel process model, we can obtain the optimal control law by solving a sparse learning convex optimization problem. Moreover, the uncertainty of the process model is taken into consideration to obtain more robust control results. A case study on FEA shows that our proposed method outperforms the current industry practice.

The three studies summarized above will be elaborated in Chapter 2, Chapter 3, and Chapter 4, respectively. We believe that these studies provide some effective control strategies for high-precision manufacturing.

CHAPTER 2. ADAPTIVE CAUTIOUS REGULARIZED RUN-TO-RUN CONTROLLER FOR LITHOGRAPHY PROCESS

2.1 Introduction

Lithography has been a major bottleneck of quality improvement in semiconductor manufacturing (Chien et al., 2014). A key challenge of the lithography process is controlling the overlay error, which is the positional error between the patterns of the photo-resist material in two neighboring layers (Park et al., 2005). As the critical dimension of the semiconductor product keeps decreasing, the requirement of overlay control becomes increasingly important to maintain high precision and a high yield to reduce the per-die cost. For example, the overlay needs to be controlled within 10nm for a critical layer in a 40nm process node. Improving the overlay control performance is critical for meeting the fab requirements and lower the cost of machine ownership (Huang et al., 2012).

Currently, run-to-run controllers are applied to reduce the overlay error during the manufacturing process (Huang et al., 2008). Run-to-run control uses the in-line data to update the settings of the automatic controller after each production run, and hence compensates the process drift and shift, and reduces process variability. In current lithography processes, the control of the overlay error is typically achieved in the following way. First, the overlay error is decomposed into three components: the error associated with the entire wafer, the common error across all fields, and the individual error of each field. The overlay error along the x axis and the y axis from each component above is then further represented by polynomials whose coefficients are adjusted by individual EWMA

controllers. In literature (Armitage et al 1988), the wafer-level component and the common error across all fields are represented by third-order polynomials, and the control through their coefficients is referred to as a higher-order process control (HOPC). The individual errors of each field along x and y axes are represented by linear functions and the control is referred to as a field-by-field control (FxFc). The combination of HOPC and FxFc is widely applied in overlay error correction.

Although HOPC and FxFc offer accurate adjustments to reduce the overlay error across the entire wafer, there are three shortcomings of the current algorithm implemented in the overlay control.

- When implementing an EWMA controller, the weight parameter λ significantly influences the estimation phase of the EWMA controller and hence impacts the performance of the EWMA controller (Castillo et al., 1997) (See Section II-C for a brief introduction of the EWMA algorithm). In current practice, the value of λ is set as a fixed value according to the properties of the disturbance patterns (Huang et al., 2008). However, due to the dynamic nature of the disturbance, the prescribed value of λ may not lead to the optimal performance of the controller. Therefore, we need to monitor the change of the system and adjust our tuning parameter λ accordingly.
- The implementation of the run-to-run controller for each coefficient relies on the parameter of the process gain, which is estimated from the offline testing data and thus is subject to estimation errors (Huang et al., 2008). However, there is no consideration of those estimation errors in the conventional EWMA controller

(Good et al., 2002), which makes the controller sensitive to the noise and estimation uncertainties in the control phase.

- In practice, the adjustable ranges of the coefficients are bounded. If some adjustment values specified by an EWMA controller are beyond the specified range of adjustment, ad-hoc methods are often used to revise the adjustment value via some prescribed transfer rules after the control recipes are calculated from all EWMA controllers. However, such ad-hoc methods may not yield optimal control performances.

In this chapter, we address those three limitations of the EWMA controller by proposing an *adaptive, cautious, and regularized* EWMA controller, which learns the λ value sequentially from the historical data, considers the uncertainties of the process gain, and incorporates the constraints for the control parameters' range. The improvements are achieved by implementing and integrating three algorithms: the adaptive algorithm, the cautious control algorithm, and the regularization algorithm upon the estimation phase and control phase of the original EWMA controller (see Figure 1). The effectiveness of our proposed method will be demonstrated in a series of simulation studies designed from real system setups.

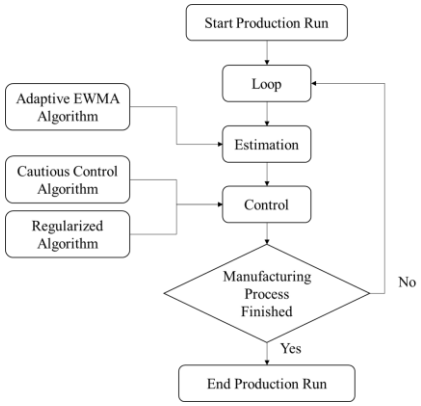


Figure 1. Proposed adaptive, cautious, and regularization control algorithm

The rest of the chapter is organized as follows. In Section 2, we provide a review of the data structure of the overlay data and the current EWMA control strategies for overlay errors. The newly developed *adaptive, cautious, and regularized* EWMA controller is presented in Section 3. Section 4 discusses the simulation settings and the results. Finally, Section 5 concludes the article.

2.2 Literature review

In this section, we provide a brief review of the lithography process, the data structure of the overlay measurements, and the existing high order polynomial control (HOPC) and field-by-field control (FxFc) schemes.

2.2.1 Lithography process and overlay measurements

Lithography is a process that generates 2-D patterns on wafer surfaces during a semiconductor manufacturing process. In this process, an optical system projects the patterns on a mask to a thin layer of photoresist material on the wafer. As the photoresist material exposed to the light, it quickly solidifies and cannot be washed away from the wafer surface, while the unexposed photoresist material can be washed away. Therefore, the patterns on the mask are transplanted to the photoresist layer, which determines the region on the wafer that will be removed in the subsequent etching process. For the process we consider, the entire wafer is comprised of m identical rectangular fields, and one chip is fabricated on each of them. In one lithography process, the wafer is processed through m times of exposure, one field at a time. After each exposure, the wafer moves horizontally to enable another time of exposure.

The major quality measurement for the lithography process is the overlay error, which represents the alignment inaccuracy between the photo resist material and the pattern of a previous layer. The measurements of overlay error are in the form of 2D vectors, denoting the relative locational difference between the pattern on the previous layer and the photoresist material. In the contemporary lithography process, the overlay measurements are taken at multiple sites within every field of the wafer to fully characterize the alignment error across the wafer. For example, the measurements on one sample wafer are shown in Figure 2 (Huang et al., 2012). In Figure 2, the grids represent the boundaries of the cells; and the vectors are the measurement 2D vector, whose value on each axis denotes the overlay error on the corresponding axis. The collection of all overlay measurements gives a sketch of the entire overlay vector field of the wafer.

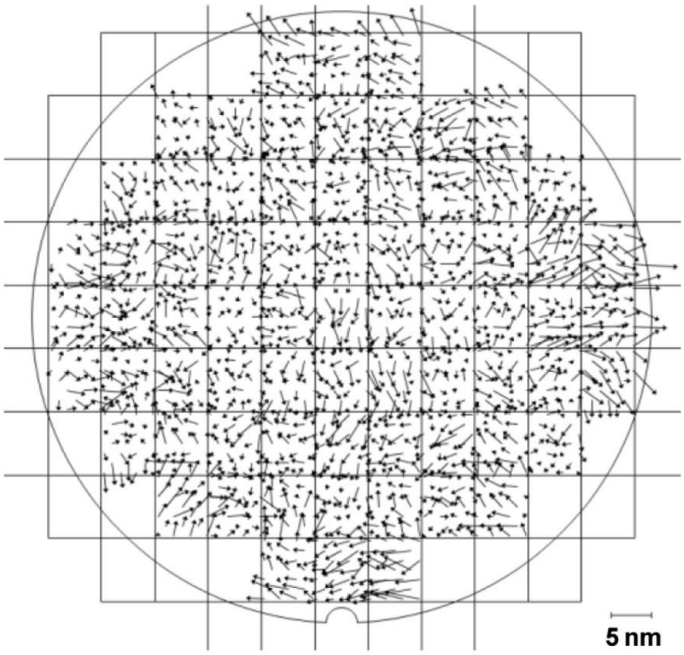


Figure 2. Illustration of the overlay measurements on a wafer (Brunner et al. 2013)

2.2.2 Overlay error modeling

In order to introduce the third-order polynomial model, we first introduce two different coordinate systems: the wafer-level coordinate system and the field-level coordinate system. As shown in Figure 3, we define a wafer-level coordinate system (X, Y) whose origin is at the center of the wafer and the X-axis is parallel to the flat edge of a wafer, and thus parallel to one edge of the die. Within every field, we also define a field-level coordinate (x, y), whose origin is the center of each field, and the x-axis is aligned with the X-axis in the wafer-level coordinate system.

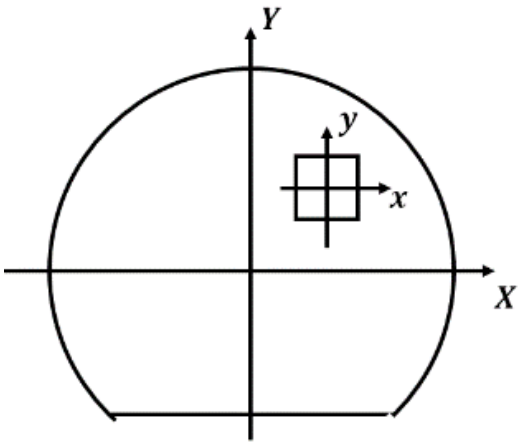


Figure 3. Illustration of the wafer coordinate system and the field coordinate system

During m times of exposures, the overlay error can be decomposed into three major sources (Huang et al., 2012):

- *The wafer-level errors*. This error source affects the overlay error across the entire wafer, such as the stage control error, the wafer distortion, etc. It can be represented by a mapping from $\mathbb{R}^2$to $\mathbb{R}^2$, $\left(F_x(X, Y), F_y(X, Y)\right)^T$, where (X, Y) denotes a wafer-

level coordinate of a measurement on the wafer, and $F_x(X,Y), F_y(X,Y)$ represent the wafer-level error along the x and y axes at this location.

- *The common field-level errors*. They represent the sources that simultaneously affect each field on a wafer, such as the error associated with the exposure system. The common field errors lead to the same overlay error pattern within each field, and thus can be represented by $\left(f_x(x,y), f_y(x,y)\right)^T$, where (x,y) is the field-level coordinate of the location.
- *The individual field-level errors*. Individual field errors are caused by the sources that individually affect the overlay of each field. For field i, the individual field level error is represented by $\left(f_{i,x}(x,y), f_{i,y}(x,y)\right)^T$.

The total overlay error is represented as the summation of the errors from these three sources. Therefore, the overlay vector $\left(\Delta_x, \Delta_y\right)$ measured at a location in field j, whose wafer-level coordinate is (X,Y), and the field coordinate is (x,y) can be represented by

$$\Delta_x(X,Y,x,y,j) = F_x(X,Y) + f_x(x,y) + f_{j,x}(x,y) \quad (1a)$$

$$\Delta_y(X,Y,x,y,j) = F_y(X,Y) + f_y(x,y) + f_{j,y}(x,y) \quad (1b)$$

With this representation, the entire overlay error field on a wafer is specified by $F_x(X,Y), F_y(X,Y); f_x(x,y), f_y(x,y);$ and $\left\{f_{j,x}(x,y), f_{j,y}(x,y)\right\}_{j=1,\dots,n}$. Here n represents the total number of fields.

The state-of-art lithography machines are equipped with HOPC and dynamic FxFc functionalities (Huang et al., 2012). The wafer-level components and the common field-level components can be represented through the third-order polynomial models as shown in Equation 2 and Equation 3. These components are adjusted through the knobs that separately control the coefficients $K_1, \dots, K_{20}$ and $k_1, \dots, k_{20}$. On the other hand, the field-by-field control is implemented to decrease the individual field-level components that are represented in linear models (Equation 4). Every coefficient $k'_{j,i}$ is controlled through an FxFc knob.

$$\begin{cases} F_x(X,Y) = K_1 + K_3X + K_5Y + K_7X^2 + K_9XY + K_{11}Y^2 + K_{13}X^3 + K_{15}X^2Y + K_{17}XY^2 + K_{19}Y^3 \\ F_y(X,Y) = K_2 + K_4X + K_6Y + K_8X^2 + K_{10}XY + K_{12}Y^2 + K_{14}X^3 + K_{16}X^2Y + K_{18}XY^2 + K_{20}Y^3 \end{cases} \quad (2)$$

$$\begin{cases} f_x(x,y) = k_1 + k_3x + k_5y + k_7x^2 + k_9xy + k_{11}y^2 + k_{13}x^3 + k_{15}x^2y + k_{17}xy^2 + k_{19}y^3 \\ f_y(x,y) = k_2 + k_4x + k_6y + k_8x^2 + k_{10}xy + k_{12}y^2 + k_{14}x^3 + k_{16}x^2y + k_{18}xy^2 + k_{20}y^3 \end{cases} \quad (3)$$

$$\begin{cases} f_{j,x}(x,y) = k'_{j,1} + k'_{j,2}x + k'_{j,3}y \\ f_{j,y}(x,y) = k'_{j,4} + k'_{j,5}x + k'_{j,6}y \end{cases} \quad (4)$$

In summary, the coefficients $K_i, k_i, i = 1, \dots, 20$ and $k'_{j,i'}, j = 1, \dots, n, i' = 1, \dots, 6$ denote the coefficients of the wafer-level variation patterns, the field-level variation patterns, and the individual variation patterns. These coefficients jointly characterize the overlay patterns on the wafer and are adjusted individually on lithography equipment.

2.2.3 Run-to-run control for overlay data

The current practice of HOPC and FxFc is to apply individual run-to-run control schemes to adjust the control coefficients K_i's, k_i's and $k'_{j,i}'$s between production runs.

Here we first review the run-to-run control schemes and then review how it is implemented in overlay control.

A run-to-run control scheme consists of three key components: a process model that specifies the relationship between the control parameter u_t and the output y_t, a control objective function, and control law. A widely used run-to-run controller in the lithography process is the EWMA controller, which is known for its simplicity and stability (Chien et al 2014). Conventionally, the EWMA controller is aimed at controlling a scalar output using a scalar control variable. Its process model is specified as

$$y_t = \beta u_t + d_t \tag{5}$$

Here $\{d_t\}$ is the process disturbance, and β is referred to as the process gain. The control objective is to minimize the mean squared error $\mathbb{E}[y_t - T]^2$, where T is the target. To achieve it, the control law of an EWMA controller consists of the following two phases

- *The estimation phase* - the disturbance is estimated at time t using the recursive EWMA scheme

$$\hat{d}_t = (1-\lambda)\hat{d}_{t-1} + \lambda(y_{t-1} - \beta u_{t-1}); \tag{6}$$

- *The control phase* - with the estimated disturbance, the control variable is selected to minimize the control objective function

$$u_t = \frac{y_t - \hat{d}_t}{\beta}. \tag{7}$$

In an EWMA controller, the weight λ plays an important role to ensure the control performance (Chang et al., 2012). The weight λ should be selected according to the

dynamics of the disturbance process $\{d_t\}$. In particular, when a disturbance follows an IMA (1, 1) model, an EWMA controller is optimal when the weight λ is the same as the IMA (1, 1) parameter.

In the current lithography process, run-to-run control is implemented for minimizing multidimensional overlay error. This is achieved by transforming the observed overlay error field to the coefficients of $\{K_i\}_{1\le i\le 20}, \{k_i\}_{1\le i\le 20}$ and $\{k'_{j,i}\}_{1\le j\le m; 1\le i\le 6}$ using the regression technique. Then, based on the calculated coefficients, the control actions for each coefficient can be derived using individual EWMA controllers. As every controller minimizes the variation of an individual coefficient, the total overlay error of the entire wafer can be reduced effectively when the knobs are adjusted according to the control law. Therefore, the current control scheme is summarized in the following procedure.

1. Solve the regression model to obtain the wafer-level coefficients $K_1, \dots, K_{20}$, common field-level coefficients $k_1, \dots, k_{20}$, and individual field-level coefficients $k'_{j,1}, \dots, k'_{j,6}; j = 1, \dots, m$.
2. Apply the EWMA control law to wafer-level coefficients $K_1, \dots, K_{20}$, common field-level coefficients $k_1, \dots, k_{20}$, and individual field-level coefficients $k'_{j,1}, \dots, k'_{j,6}; j = 1, \dots, m$ respectively.

2.3 Adaptive cautious regularized run-to-run controller

In this section, we will propose a new run-to-run controller that (1) is able to adjust the tuning parameter λ adaptively during the run-to-run control process, (2) takes the uncertainty of process gain into consideration, and (3) performs the run-to-run control

effectively when the adjustable range of the knobs are bounded. We will discuss each of those three capabilities in the subsections below, and then integrates these capabilities in our proposed controller.

2.3.1 Adaptive control strategy

In the existing EWMA controller, the parameter λ is typically set as a fixed value, which does not follow the change of process dynamics over time. To address this issue, we develop an adaptive control strategy by automatically adjusting parameter λ through learning its value from the historical data (Bollen et al., 2014). The idea is to minimize the prediction error over the last N data points within a moving window by setting the value of λ_t appropriately. In order to do so, we solve the optimization problem

$$\lambda_t = \arg\min_{\lambda} \sum_{i=1}^{N} \left(d_{t-i} - \hat{d}_{t-i}(\lambda)\right)^2, \tag{8}$$

where $\hat{d}_{t-i}(\lambda)$ is recursively calculated with

$$\hat{d}_{t-i}(\lambda) = \lambda d_{t-i-1} + (1-\lambda)\hat{d}_{t-i-1}(\lambda); \hat{d}_1(\lambda) = 0,$$

and $d_j = y_j - \beta u_j$ for all $j = 1, \dots, t-1$. The minimization problem (8) is solved numerically after each production run. With this online adjustment strategy, the procedure for implementing the adaptive control strategy involves three steps:

1. Fix a prescribed λ_0, and employ the conventional EWMA controller for the first N wafers.
2. At time $t > N$, we use the data points obtained at time $t-1, \dots, t-N$ to calculate λ_t, according to Equation 8.

3. After obtaining λ_t, it is applied to the EWMA controller and the control law u_t is calculated from Equations 6 and 7 using $\lambda = \lambda_t$.

2.3.2 Cautious control strategy

In an EWMA controller, the relationship among the control action u_t, disturbance d_t and the response y_t is specified as

$$y_{t+1} = \beta u_t + d_t.$$

The coefficient β is the process gain that represents the effect of the control variable u_t on the response y_{t+1}. Conventionally, β is assumed to be a known and fixed value. However, in practice, the value of β is unknown and thus must be estimated from a calibration process before the controller is set up. Therefore, the estimation of $\hat{\beta}$ is typically different from the true process gain β. Due to the nature of the calibration process, we may assume that the posterior distribution of the true process gain $\beta|\hat{\beta}$ given the estimated process gain $\hat{\beta}$ follows a Normal distribution $N\left(0, \sigma_{\hat{\beta}}^2\right)$, and the variance $\sigma_{\hat{\beta}}^2$ is obtained from the calibration procedure (Apley et al., 2004) (Bar-Shalom et al., 1981). Based on this, the cautious control concept (Apley et al., 2004) (Bar-Shalom et al., 1981) can be implemented for the run-to-run control process. In particular, the control variable u_t that minimizes $J(u_t) = \mathbb{E}[(y_t - T)^2]$ can be obtained as

$$u_t = \frac{\hat{\beta}(T - d_t)}{\hat{\beta}^2 + \sigma_{\hat{\beta}}^2}. \tag{9}$$

The derivation Equation 9 is given in Appendix A.1.

As can be seen from Equation 9, the control law u_t is not only a function of the estimation value of the gain β, but also a function of the variance of the estimation $\sigma_{\hat{\beta}}^2$. When the estimation error is small, the value $\sigma_{\hat{\beta}}^2$ is small and the control law u_t is similar to the conventional EWMA control. However, when the estimation error is large, the cautious control law in Equation 9 will lead to a smaller control value, thus improve the robustness of the control to the estimation error. The cautious control method can be directly applied to adjust the parameters $K_1, \ldots, K_{20}$; $k_1, \ldots, k_{20}$; and $k'_{j,1}, \ldots, k'_{j,6}$ for $j = 1, \ldots, m$ respectively.

2.3.3 Regularized control strategy

In this section, we propose a strategy that improves the existing control method when the range of the control variable is bounded by the machine specifications.

Let the control variables associated with K_i, k_i and $k'_{j,i}$ be U_i, u_i and $u'_{j,i}$ respectively for all i and j properly defined, and let the corresponding adjustable ranges be $\left[\underline{L}(U_i), \overline{L}(U_i)\right], \left[\underline{L}(u_i), \overline{L}(u_i)\right]$ and $\left[\underline{L}(u'_{j,i}), \overline{L}(u'_{j,i})\right]$. Let the i^{th} wafer-level and the field-level coefficients at time t be $K_{i,t}$ and $k_{i,t}$, and the corresponding disturbance terms are $D_{i,t}$, $d_{i,t}$ for the wafer-level and the common field-level respectively. From the process model, the control coefficients at time $t+1$ can be represented as

$$K_{i,t+1} = U_{i,t} + D_{i,t}, i = 1, \ldots, 20; \tag{10a}$$

$$k_{i,t+1} = u_{i,t} + d_{i,t}, i = 1, \ldots, 20. \tag{10b}$$

At each time t, these control variables should be set to the value specified by their individual controllers. However, this requirement cannot be achieved if these values are out of the adjustable range. In such a case, a simple way is to make the adjustment to the value within the range and closest to the value specified by the control law. However, such a method does not achieve the best control performance in minimizing the overlay error, as the effect of an out-of-range coefficient sometimes can be carried over to another adjustable coefficient within the range. For example, if k_3, the coefficient that corresponds to the x term, is already at the upper limit $\overline{L}(u_3)$ and needs to be further increased, we can resort to increase k_{13} (i.e. the coefficient of the x^3 term) if possible, as it has a similar effect to that of k_3. Hence, compared with using individual univariate EWMA controller to calculate each control variable individually and independently, adjusting multiple control variables jointly using a multivariate controller may achieve better performance. Therefore, we propose a regularization strategy that determines the control variables by minimizing the magnitude of the overlay error. Here the word "regularization" means to constrain all controllable coefficients within their individual ranges when solving the minimization problem.

First, we represent the total overlay magnitudes by a function of the coefficients. Recall that in Equation 1, the total error is decomposed into wafer-level, common field-level, and independent field-level errors. Among them, the wafer-level and the common field-level components are adjusted by knobs and are affected by a limited adjustment range. Due to there is no bound for the individual field-level error, we just set all of the individual field-level error to zero for simplicity. Therefore, the total overlay error is simplified to only consist of the wafer-level and the common field-level components. In

the control law derivation, we represent the wafer-level and the common field-level overlay error components into a matrix form with a high order polynomial basis:

$$\mathbf{F}_x = \mathbf{W}_F\mathbf{K}_x, \mathbf{F}_y = \mathbf{W}_F\mathbf{K}_y; \quad (11a)$$

$$\mathbf{f}_x = \mathbf{W}_f\mathbf{k}_x, \mathbf{f}_y = \mathbf{W}_f\mathbf{k}_y; \quad (11b)$$

In Equations 11a, $\mathbf{F}_x$ and $\mathbf{F}_y$ represent the wafer-level overlay error on the x axis and the y axis respectively. Each of them is a vector of dimension $n = \sum_{i=1}^{m} n_i$, where n is the number overlay measurement sites on the wafer. Here $\mathbf{K}_x = (K_1, K_3 \dots, K_{19})'$ and $\mathbf{K}_y = (K_2, K_4 \dots, K_{20})'$ are the coefficient vectors, and

$$\mathbf{W}_F = \begin{bmatrix} 1 & X_1 & \cdots & X_1Y_1^2 & Y_1^3 \\ 1 & X_2 & & X_2Y_2^2 & Y_2^3 \\ & \vdots & \ddots & \vdots & \\ 1 & X_{n-1} & \cdots & X_{n-1}Y_{n-1}^2 & Y_{n-1}^3 \\ 1 & X_n & & X_nY_n^2 & Y_n^3 \end{bmatrix},$$

is the regressor matrix, where (X_i, Y_i) represents the wafer coordinates of the measurement site i. In Equation 11b, $(\mathbf{f}_x, \mathbf{f}_y)'$ represents the common field-level overlay components at all sites on the wafer, and $\mathbf{k}_x, \mathbf{k}_y$ represents the corresponding field-level coefficients. The matrix $\mathbf{W}_f$ corresponds to $\mathbf{W}_F$, through replacing the wafer coordinates (X, Y) of every measurement site with its field coordinates (x, y).

To derive the objective function, we represent the sum-of-square overlay error caused by wafer-level and the common-field level components using control variables. With the notations introduced above, the sum-of-square overlay error equals to $(\mathbf{F}_x + \mathbf{f}_x)^{\boldsymbol{T}}(\mathbf{F}_x + \mathbf{f}_x) + (\mathbf{F}_y + \mathbf{f}_y)^{\boldsymbol{T}}(\mathbf{F}_y + \mathbf{f}_y)$.We use the residual of wafer-level regression as the dependent variable of field-level regression, Therefore, the field-level components

$\mathbf{f}_x, \mathbf{f}_y$ are approximately orthogonal to $\mathbf{F}_x$ and $\mathbf{F}_y$. We have $\mathbf{f}_x^T\mathbf{F}_x \approx 0$ and $\mathbf{f}_y^T\mathbf{F}_y \approx 0$, so that this representation is equivalent with

$$\text{minimize } \{\mathbf{F}_x^T\mathbf{F}_x + \mathbf{F}_y^T\mathbf{F}_y + \mathbf{f}_x^T\mathbf{f}_x + \mathbf{f}_y^T\mathbf{f}_y\} \tag{12}$$

Substitute Equation 11 into Equation 12, it is transformed to

$$\text{minimize } \{\mathbf{K}_x^T\mathbf{W}_F^T\mathbf{W}_F\mathbf{K}_x + \mathbf{K}_y^T\mathbf{W}_F^T\mathbf{W}_F\mathbf{K}_y + \mathbf{k}_x^T\mathbf{W}_f^T\mathbf{W}_f\mathbf{k}_x + \mathbf{k}_y^T\mathbf{W}_f^T\mathbf{W}_f\mathbf{k}_y\} \tag{13}$$

After obtaining the prediction of the disturbances corresponding to the coefficients, $\{U_{i,t}\}, \{u_{i,t}\}$ can be calculated to minimize the objective function in problem 14

$$\begin{aligned}\text{minimize } &(\mathbf{U}_{x,t}^T + \widehat{\mathbf{D}}_x^T)\mathbf{W}_F^T\mathbf{W}_F(\mathbf{U}_{x,t} + \widehat{\mathbf{D}}_x) + (\mathbf{U}_{y,t}^T + \widehat{\mathbf{D}}_y^T)\mathbf{W}_F^T\mathbf{W}_F(\mathbf{U}_{y,t} + \widehat{\mathbf{D}}_y) \\ &+ (\mathbf{u}_{x,t}^T + \hat{\mathbf{d}}_x^T)\mathbf{W}_f^T\mathbf{W}_f(\mathbf{u}_{x,t} + \hat{\mathbf{d}}_x) + (\mathbf{u}_{y,t}^T + \hat{\mathbf{d}}_y^T)\mathbf{W}_f^T\mathbf{W}_f(\mathbf{u}_{y,t} + \hat{\mathbf{d}}_y),\end{aligned} \tag{14}$$

with each parameter in its respective adjustable range. Here $\hat{\mathbf{d}}_{x,t}, \hat{\mathbf{d}}_{y,t}, \widehat{\mathbf{D}}_{x,t}, \widehat{\mathbf{D}}_{y,t}$ and $\mathbf{u}_{x,t}, \mathbf{u}_{y,t}, \mathbf{U}_{x,t}, \mathbf{U}_{y,t}$ are vectors comprised of the predicted disturbances and control variables, similar to the coefficient matrices $\mathbf{k}_x, \mathbf{k}_y, \mathbf{K}_x, \mathbf{K}_y$.

Observing problem 14, we find that it can be separated into four individual optimization problems, whose objective functions are the four parts, and whose control variables are $\mathbf{U}_{x,t}, \mathbf{U}_{y,t}$, $\mathbf{u}_{x,t}$ and $\mathbf{u}_{y,t}$ respectively. For example, the solution to the problem associated with $\mathbf{U}_{x,t}$ is

$$\text{minimize } (\mathbf{U}_{x,t}^T + \widehat{\mathbf{D}}_x^T)\mathbf{W}_F^T\mathbf{W}_F(\mathbf{U}_{x,t} + \widehat{\mathbf{D}}_x) \tag{15}$$

$$\text{subject to } \underline{\mathbf{L}}(\mathbf{U}_x) \leq \mathbf{U}_{x,t} \leq \overline{\mathbf{L}}(\mathbf{U}_x),$$

and the other three problems have similar forms. Problem 15 is a bounded quadratic programming problem, which can be solved effectively by alternating direction method of multiplier (ADMM) (Boyd et al., 2012). A detailed procedure of the algorithm is given in Appendix A.2. After the parameters $\mathbf{u}_{x,t}, \mathbf{u}_{y,t}, \mathbf{U}_{x,t}, \mathbf{U}_{y,t}$ are solved, the control laws subject to the restrictions can be obtained. In summary, the overall procedure to implement the regularized controller include:

1. Obtain the estimation of the disturbance term with the EWMA scheme $\widehat{D}_{1:10}(t)$, $\widehat{D}_{11:20}(t), \hat{d}_{1:10}(t)$ and $\hat{d}_{11:20}(t)$. For example, $\widehat{D}_{1:10}(t)$ is calculated from $\widehat{D}_{1:10}(t) = \lambda\big(K_{1:10}(t) - U_{1:10}(t-1)\big) + (1-\lambda)\widehat{D}_{1:10}(t-1)$.
2. Solve four bounded quadratic programming problems that are in the form of Problem 15 using the ADMM algorithm in Appendix A.2, to obtain the new control law $U_{1:10}(t)$, $U_{11:20}(t)$,$u_{1:10}(t)$,and $u_{11:20}(t)$ respectively.

2.3.4 The adaptive cautious regularized EWMA controller

In this section, we unify the control strategies discussed in the previous three subsections and propose an adaptive cautious regularized EWMA control scheme. First, we combine the EWMA cautious strategy discussed in Section 2.3.2 with the regularized EWMA control strategy in Section 2.3.3 as follows. As shown in Appendix A.3, the cautious control strategy of univariate controller calculates the control law by minimizing $J(u_t)$, the expected squared error subject to the uncertainty of the process gain. By applying the idea of cautious control to the regularized control, we change our objective function from the summation of squared prediction error in Problem 16 to the expected sum of squared prediction error:

$$\begin{aligned}\text{minimize } \mathrm{E}\Big\{&\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_X\big)\mathbf{U}_{x,t}^{\boldsymbol{T}}+\widehat{\mathbf{D}}_x^{\boldsymbol{T}}\Big)\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_X\big)\mathbf{U}_{x,t}+\widehat{\mathbf{D}}_x\Big)\\&+\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_Y\big)\mathbf{U}_{y,t}^{\boldsymbol{T}}+\widehat{\mathbf{D}}_y^{\boldsymbol{T}}\Big)\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_Y\big)\mathbf{U}_{y,t}+\widehat{\mathbf{D}}_y\Big)\Big\}\\&+\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_x\big)\mathbf{u}_{x,t}^{\boldsymbol{T}}+\hat{\mathbf{d}}_x^{\boldsymbol{T}}\Big)\mathbf{W}_f^{\boldsymbol{T}}\mathbf{W}_f\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_x\big)\mathbf{u}_{x,t}+\hat{\mathbf{d}}_x\Big)\\&+\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_y\big)\mathbf{u}_{y,t}^{\boldsymbol{T}}+\hat{\mathbf{d}}_y^{\boldsymbol{T}}\Big)\mathbf{W}_f^{\boldsymbol{T}}\mathbf{W}_f\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_y\big)\mathbf{u}_{y,t}+\hat{\mathbf{d}}_y\Big),\end{aligned} \tag{16}$$

which can be further decomposed into four optimization problems. Here $\widetilde{\mathbf{B}}_X, \widetilde{\mathbf{B}}_Y, \widetilde{\mathbf{B}}_x$ and $\widetilde{\mathbf{B}}_y$ are 10×10 diagonal matrices, whose (i, i) the element represents the error of the process gain for the corresponding coefficient in K_x, K_y, k_x and k_y, following Normal distributions. The term related to $\mathbf{U}_{x,t}$ then changes from Problem 15 to the following Problem 17:

$$\begin{aligned}&\text{minimize } \mathbb{E}\Big\{\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_X\big)\mathbf{U}_{x,t}^{\boldsymbol{T}}+\widehat{\mathbf{D}}_x^{\boldsymbol{T}}\Big)\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\Big(\big(\mathbf{I}+\widetilde{\mathbf{B}}_X\big)\mathbf{U}_{x,t}+\widehat{\mathbf{D}}_x\Big)\Big\}\\&\text{subject to } \underline{\mathbf{L}}(\mathbf{U}_x) \le \mathbf{U}_{x,t} \le \overline{\mathbf{L}}(\mathbf{U}_x),\end{aligned} \tag{17}$$

After the mathematical transformation shown in Appendix A.3, the following final objective function can be obtained, as shown in Equation 18

$$\text{minimize } \{\mathbf{U}_{x,t}^{T}\Big(\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F+\mathbb{E}\big(\widetilde{\mathbf{B}}_X\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\widetilde{\mathbf{B}}_X\big)\Big)\mathbf{U}_{\boldsymbol{x,t}}+\mathbf{2}\widehat{\mathbf{D}}_{\boldsymbol{x}}^{\boldsymbol{T}}\mathbf{W}_{\boldsymbol{F}}^{\boldsymbol{T}}\mathbf{W}_{\boldsymbol{F}}\mathbf{U}_{\boldsymbol{x,t}}\} \tag{18}$$

After the cautious control strategy and the regularized control strategy are combined, the adaptive control strategy can then be applied on top of this combined controller. The general framework is shown in Figure 4. After obtaining the new measurement data, we first use the adaptive EWMA algorithm to calculate λ_t, the best EWMA parameter based on the past observations. Then, the cautious regularized controller

can be applied to calculate the control law based on the optimized λ_t. The overall adaptive cautious regularized controller is shown in Figure 4.

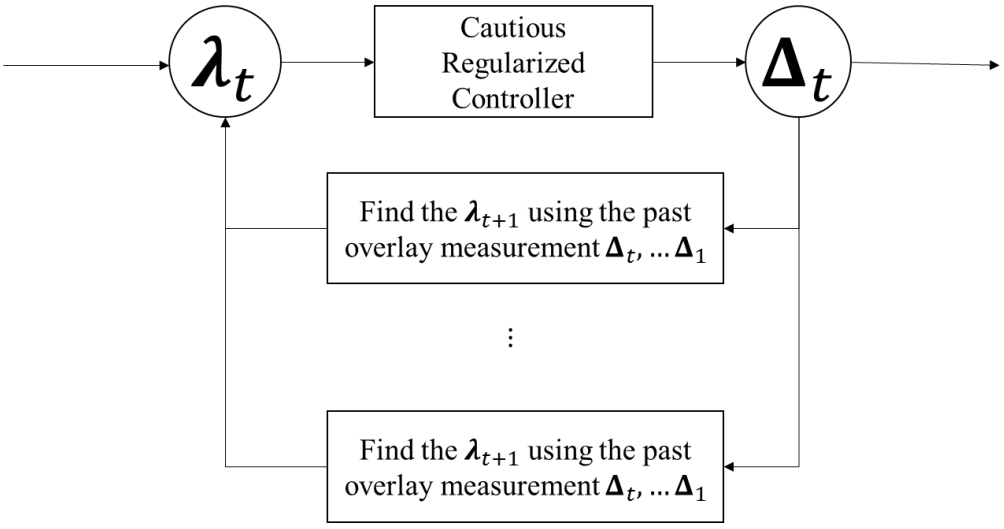


Figure 4. Flowchart of adaptive cautious regularized controller

In the figure, $\boldsymbol{\Delta}_t$ represents all overlay vectors obtained from the wafer t; and $\boldsymbol{\lambda}_t$ represents the vector contains all tuning parameter $\lambda_1, \ldots, \lambda_{40}$. Here, $\lambda_{1,t}, \ldots, \lambda_{10,t}$ are the tuning parameters for the wafer-level error on the X-axis at time t; $\lambda_{11,t}, \ldots, \lambda_{20,t}$ are the tuning parameters for the wafer-level error on Y-axis at time t. Similarly, $\lambda_{21,t}, .., \lambda_{30,t}$ and $\lambda_{31,t}, \ldots, \lambda_{40,t}$ represent the tuning parameters for common field-level on the x-axis and the y axis respectively. By following the framework shown in Figure 4, the three strategies proposed in Section 2.3.1, 2.3.2, and 2.3.3 are combined for controlling a lithography system. As a result, the developed adaptive cautious regularized control scheme is able to learn λ adaptively, and takes both the model uncertainty and adjustable range of the control variables into consideration. The effectiveness of this framework will be demonstrated in the simulation study in the next section.

2.4 Simulation study

In this simulation study, we first demonstrate the effectiveness of the cautious control strategy, the adaptive control strategy, and the regularized control strategy respectively by comparing each of them with the conventional EWMA control scheme (Section IV-A). In Section IV-B, we demonstrate the effectiveness of the controller when these three strategies are combined: we first validate the effectiveness of the adaptive cautious controller when adaptive control strategy and the cautious control strategy are integrated, and then validate the effectiveness of the adaptive cautious regularized controller when all three strategies are integrated.

2.4.1 Illustration and validation of the newly proposed individual control strategies

2.4.1.1 Cautious control strategy

We first compare the cautious control strategy with the conventional EWMA control. First, we assume that the process gain $\sigma_{\tilde{\beta}}^2$ is a positive constant and compares the control performance of the cautious controller with the conventional EWMA controller, when λ is set to certain fixed values. Then, we fix λ to the optimal value according to the temporal correlation of the process and investigate how $\sigma_{\tilde{\beta}}^2$, the measure of the uncertainty for the process gain β, affects the control performance.

Case I. The comparison between the cautious controller and conventional EWMA controller under different values of λ.

We consider a case that both the control input and the process output are univariate. The mean of β is set as 2, and $\sigma_{\tilde{\beta}}^2$ is set to 0.5. The performance of the cautious

controller and the conventional EWMA controller when λ takes different values between 0.05 and 1 is illustrated in Figure 5. Here, the vertical axis denotes the mean-squared error and the horizontal axis denotes the value of λ that we used in the controller. From this figure, we can observe that the mean-squared error of the cautious controller (denoted by the red curve) is always smaller than the mean-squared error of the conventional EWMA controller (denoted by the green curve), regardless of the value of λ. An improvement rate of approximately 5% is achieved.

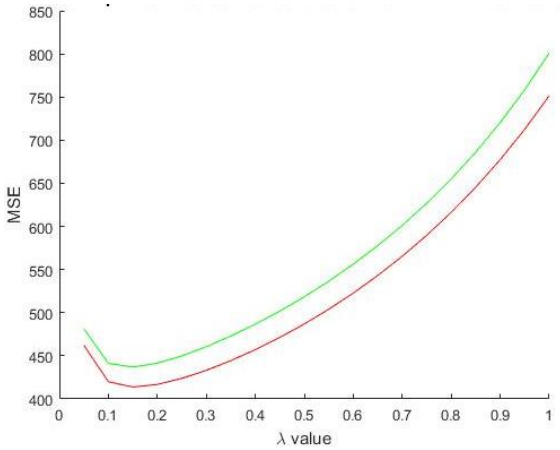


Figure 5. Comparison of the cautious controller and EWMA controller

Case II. The comparison between the cautious controller and conventional controller under different values of $\sigma_{\tilde{\beta}}^2$.

We then evaluate the performance of the cautious controller when the level of model uncertainty varies. Here, we simulate the disturbance from an IMA model with parameter λ =0.3, and set the parameter of the EWMA controller with the same value. We adjust the standard deviation of the process gain $\sigma_{\tilde{\beta}}$ from $0.05\hat{\beta}$ to $0.5\hat{\beta}$, where $\hat{\beta}$ is 1, denoting that the mean of the posterior of β is 1. The improvement rate of the

control performance is defined as the percentage decreasing of the mean squared error of the new controller (MSE_1) from that of the conventional EWMA controller (MSE_0),

$$\text{Improvement rate} = \left(1 - \frac{MSE_1}{MSE_0}\right) \times 100\%.$$

In Figure 6, the horizontal axis denotes the value of $\frac{\sigma_{\tilde{\beta}}}{\hat{\beta}}$ and the vertical axis denotes the improvement rate. We can find that the improvement rate becomes larger as $\sigma_{\tilde{\beta}}$ increases, and that there is always an improvement of performance when implementing the cautious control strategy. Furthermore, the improvement rate becomes larger as $\sigma_{\tilde{\beta}}$ increases. The improvement of the control rate is not significant when $\sigma_{\tilde{\beta}}$ is negligible compared with $\hat{\beta}$, but the improvement become significant when $\sigma_{\tilde{\beta}}$ is greater than $0.25\hat{\beta}$.

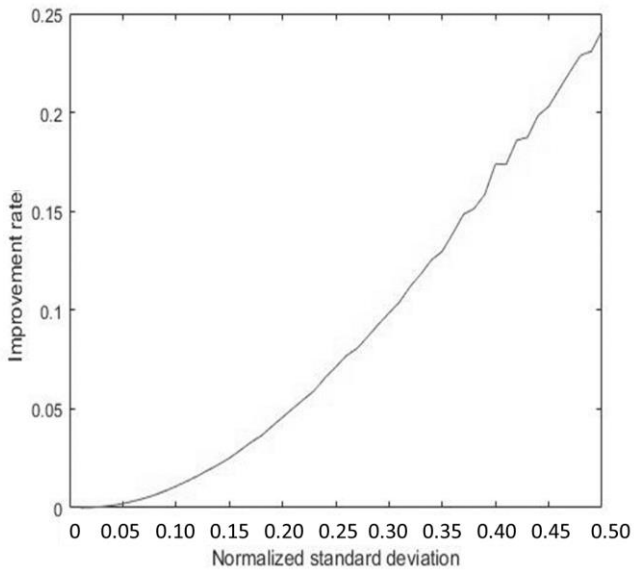


Figure 6. Percentage of improvement with different sigma/mean of intercept

2.4.1.2 Adaptive control strategy

In this study, we compare the controller implementing the adaptive control strategy with the conventional EWMA controller for a univariate response. We first simulate the disturbance of a univariate response from an IMA (1, 1) series,

$$d_t = d_{t-1} + \epsilon_t - (1-\theta)\epsilon_{t-1} \quad \epsilon_t \sim N(0, \sigma_\epsilon^2),\ \theta \in [0,1]. \tag{19}$$

and compare the performance of EWMA controllers with or without the application of adaptive EWMA strategy. Then, we simulate the disturbance from an IMA model whose parameter θ is not constant and repeat the comparison procedure.

Case 1. When the dynamics of the disturbance do not change.

We generate the disturbance from an IMA model according to Equation 19, where θ stays at 0.3 during 3000 runs. The mean of MSE of the new controller is 1.6% larger than the conventional EWMA controller and the variance of the MSE is 1.7% smaller than the conventional EWMA controller.

Case 2. The temporal correlation of the disturbance changes over time.

In this study, we change the value of θ to reflect the variation of the process's temporal correlation when the controller is applied. In particular, we generate the disturbance by setting $\theta = 0.3$ when $1 \le t \le 1000$, $\theta = 0.6$ when $1001 \le t \le 2000$, and $\theta = 0.3$ when $2001 \le t \le 3000$. The mean of MSE of the responses corresponding to the adaptive controller is 1.6% less than that of the conventional EWMA controller.

2.4.1.3 Regularized control strategy

Finally, we compare the controller with the regularization strategy with the conventional EWMA controller, when they are applied to reduce the overlay error measured from the entire wafer. We assume that there are 113 fields on a wafer, and the location of the measurements is illustrated in Figure 2. The overlay vector field of the entire wafer is generated from a simulator endorsed by a company.

In our comparison study, we test the performance of the conventional EWMA controller and the one with the regularized strategy under three settings:

(a) All $K_1, \ldots, K_{20}$ and $k_1, \ldots, k_{20}$ need to be greater than -10,000 and smaller than 10,000;

(b) All $K_1, \ldots, K_{20}$ and $k_1, \ldots, k_{20}$ need to be greater than -10 and smaller than 10;

(c) All $K_1, \ldots, K_{20}$ and $k_1, \ldots, k_{20}$ need to be greater than -1 and smaller than 1.

When the conventional EWMA controller is used and a calculated control recipe is out of the specified bound, we simply set that recipe to the closest boundary value without changing the recipes for other controllers.

The simulation was performed 3 times. For each time, 3000 wafers are generated from the simulator. The improvement rates of the controllers with regularized strategy over the conventional EWMA controller are summarized in Table 1.

Table 1. Improvement rate of the controller with regularized strategy

Improvement rate (%)	Setting (a)	Setting (b)	Setting (c)
1st replication	0	20.92	13
2nd replication	0	10.01	27.31
3rd replication	0	2.12	13.56
Mean	0	11.02	17.96

From the results in setting (a), we can see that the bounds of control variables are sufficiently wide so that they have no effect in limiting the control recipes. In such a setting, the regularized controller performs the same as the conventional EWMA controller. When the range limits of control variables are added, the controller with the regularization strategy will always perform better than the conventional EWMA controller. Under setting (b), the mean improvement rate is 11.02%; and under setting (c), the mean improvement rate increases to 17.96%. It shows that when the bounds become tighter, a higher improvement rate will be obtained.

2.4.2 Validation of the controllers combining three control strategies

In this subsection, we first demonstrate the effectiveness of the adaptive cautious controller, which integrates the adaptive control strategy and the cautious control strategy. This controller is used when the bounds of the control parameters are very wide. Then, we demonstrate the effectiveness of the adaptive cautious regularized controller. When the bounds of the control parameters are narrow, the adaptive cautious regularized controller should be applied.

2.4.2.1 The effectiveness of the adaptive cautious controller

In this section, we compare the effectiveness of the adaptive cautious controller with the conventional EWMA controller when they are used for overlay control. The overlay data sets are generated from using the same simulator as we validate the regularized control strategy in Section 2.4.1.3. For both the conventional EWMA controller and the cautious controller, the value of the parameter λ is set to be 0.3. The value of λ for the

adaptive EWMA controller and the adaptive cautious controller, instead, are learned from the historical data.

In this comparison study, we calculate μ (the average length of the overlay vectors) and σ (the standard deviation of all overlay vectors' lengths) from the overlay measurements on each wafer. Then, we use $\mu + 3\,\sigma$ to represent the performance of the controller. We compare the performance of four schemes under the following situations:

(a) The process gain has a certain level of uncertainty. In particular, we assume that the process gain $\beta \sim N(1,0.01)$.

(b) The temporal correlation of the processes varies over time. In particular, we assume that the disturbance terms for parameter $K_1, \dots, K_{20}$; $k_1, \dots, k_{20}$; and $k_{j,1}^{'}, \dots, k_{j,20}^{'}$ always follow an IMA (1, 1) model, and the change of the dynamics of the process is reflected on the parameter θ. Denote the coefficient of the IMA model for parameter K_i at time t by $\theta_{K_i,t}$, and we generate $\theta_{K_i,t}$ from the following model (which is the same for the $\theta_{k_i,t}$ and $\theta_{k_{i,j}^{'},t}$):

$$\theta_{K_i,t} = \mathrm{frac}\left(\theta_{K_i} + ct + a\varepsilon_{K_i,t}\right).$$

Here, $\theta_{K_i} + ct$ denotes the mean value of $\theta_{K_i,t}$ during the process, where θ_{K_i} is a randomly selected number between 0 and 1. The parameter c denotes the slope of the change of θ, and $\epsilon_{K_i,t} \sim N(0, \sigma_k^2)$ denotes a random variation. $\mathrm{frac}(\cdot)$ represents the fractional part of $\theta_{K_i} + ct + a\varepsilon_{K_i,t}$, which avoids the IMA parameter exceeding 1.

In our simulation study, the parameter (a, c) are selected from four combinations: (1) $a = 0.01, c = 0$; (2) $a = 0.01$, $c = 0.01$; (3) $a = 0.5$, $c = 0$; and (4) $a = 0.5$, $c =$

0.01. In cases (1) and (3), there is no drift of the temporal correlation within the process. In cases (2) and (4), θ drifts periodically. In cases (1) and (2), the uncertainty of θ is small, whereas in cases (3) and (4) the uncertainty of θ is large.

The comparison results under these four situations are summarized in Table 2. In this comparison study, the number within each cell represents the percentage of improvement over the conventional EWMA controller.

Table 2. Comparison of four types of controller

Cases	Conventional EWMA controller	Cautious controller	Adaptive controller	Adaptive cautious controller
(1) No drifting, small variance of λ	0	2.03%	10.56%	11.75%
(2) Drifting exists, small variance of λ	0	2.04%	6.38%	8.15%
(3) No drifting, large variance of λ	0	1.90%	3.63%	5.20%
(4) Drifting exists, large variance of λ	0	2.11%	8.00%	9.77%

From Table 2, we can see that the adaptive cautious EWMA controller always has the best performance, as it demonstrates the largest amount of improvement within four controllers under all four simulation settings.

2.4.2.2 Simulation of adaptive cautious regularized controller

In this section, we validate the effectiveness of the adaptive cautious regularized controller comparing with the conventional EWMA controller. Like the simulation study in Section 2.4.1.3, we test our methods concerning three different widths of the coefficient bounds, as specified in cases (a), (b), and (c) in Section 2.4.1.3. In the meantime, we also test our methods under five levels of the process gain uncertainties. In particular, the standard deviations of the process gain are 0.1, 0.2, 0.3, 0.4, and 0.5 respectively. Our

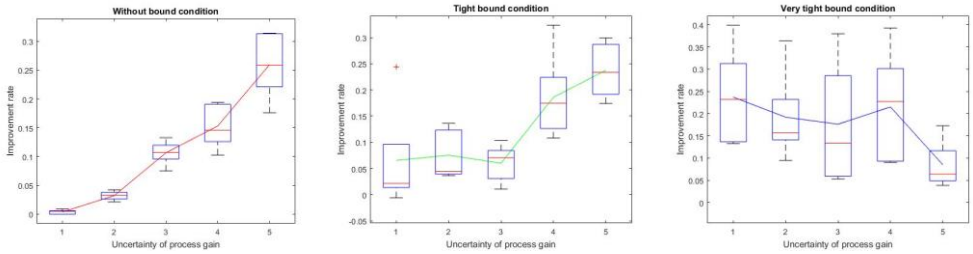
simulation has been replicated five times, and the improvement rates of the adaptive cautious regularized controller over the conventional multiple EWMA controller are summarized in Table 3.

Table 3. Improvement rates of the adaptive cautious regularized EWMA controller for different widths of the bounds and different uncertainties of the process gain

	Bounds	$\sigma_{\hat{\beta}}$				
		0.1	0.2	0.3	0.4	0.5
1st replication	(a)	0.0049	0.0329	0.0749	0.1026	0.1761
	(b)	-0.006	0.0365	0.0705	0.1082	0.1975
	(c)	0.1325	0.1881	0.3795	0.0939	0.0519
2nd replication	(a)	0.0049	0.0213	0.1329	0.1457	0.2585
	(b)	0.2441	0.0404	0.0377	0.3237	0.1741
	(c)	0.2321	0.3638	0.1337	0.3925	0.0386
3rd replication	(a)	0	0.0366	0.1152	0.1898	0.3139
	(b)	0.0216	0.0443	0.0105	0.1908	0.2831
	(c)	0.3989	0.1568	0.0530	0.0904	0.0980
4th replication	(a)	0.0091	0.0422	0.1027	0.1336	0.2356
	(b)	0.0470	0.1193	0.1033	0.1745	0.2337
	(c)	0.2840	0.1563	0.0609	0.2271	0.0639
5th replication	(a)	0	0.0280	0.1070	0.1938	0.3127
	(b)	0.0201	0.1364	0.0780	0.1326	0.2994
	(c)	0.1380	0.0946	0.2539	0.2710	0.1725

From Table 3, we can see that when there are completely no bounds and the bounds are moderately tight, the improvement rate will become larger and larger as the uncertainty of the process gain increase. However, when the bounds are very tight, the improvement

rate will decrease as the model uncertainty of the process gain increase. The result is summarized in Figure 7.



(a)Without bound (b) Tight bound (c) Very-tight bound

Figure 7. Improvement rate of the adaptive cautious bound controller under different conditions

For Figure 7 (a), (b), and (c), the x-axis denotes the duplication, the y axis denotes the improvement rate. The (a), (b), and (c) represent three different conditions, which are without bound condition, tight bound condition, and very tightly bound condition.

2.5 Conclusion

In this chapter, we identified three deficiencies of the existing run-to-run control algorithm for the HOPC and FxFc systems. Addressing these deficiencies, we proposed three control strategies: (1) an adaptive control strategy that adjusts the EWMA parameters based on the historical measurement and thereby adapts to the change of temporal correlation; (2) a cautious control strategy that takes the uncertainty of the process gain into consideration; and (3) a regularized control strategy that is able to find the best control recipes when they are restricted to bounded values. These strategies are then combined into a unified framework as an adaptive cautious regularized control scheme. The simulation study demonstrates the effectiveness of each control strategy and validates that the adaptive cautious regularized control method significantly improves the conventional EWMA

control strategy in terms of the $\mu + 3\sigma$ performance index, which is widely used in semiconductor industry.

CHAPTER 3. IMAGE-BASED FEEDBACK CONTROL USING TENSOR ANALYSIS

3.1 Introduction

System control has been widely used in a variety of manufacturing systems to target processes and reduce their variability. The main objective of a control procedure is to set control variables such that the deviation of the response/quality measure from its target is minimized. In many cases, the response variable is in the form of images or, in general, high-dimensional (HD) tensors. For example, in semiconductor manufacturing, the overlay errors of a silicon wafer, an important product quality measure, are represented by an image. In this manufacturing process, wafers are processed in multiple stages including lithography, etching, and many more before becoming final products. One of the most critical stages in this process is lithography, which projects a pre-designed pattern onto the photoresist material on the wafer through a lithography optical system. Since the lithography process is conducted layer by layer, there are some pattern misalignments between two adjacent layers, which are defined as overlay errors. An example of an overlay image is given in Fig.3. They are often caused by operation-induced stress variations during layer deposition, and/or lithographic patterning that distorts the wafer shape. Therefore, they can be controlled by adjusting the settings of the lithography machine including the wafer position, and lens height. In particular, the wafer position influences the global (wafer-level) overlay error, while lens height influences the local (field-level) overlay error. As the setting of the lithography machine changes, the global and local overlay errors will change accordingly. A proper control strategy should find the optimal

setting of the machine that leads to the minimum overlay error for the next wafer, based on the information of past overlay images and the machine settings. Other application examples in which image-based process control is important, include hot rolling (Yan et al., 2015), fuselage assembly (Yue et al., 2018), and additive manufacturing (Liu et al., 2019).

In the process control literature, there is some research on the multivariate control scheme for multi-stream signals and images. Tseng et al., (2002) proposed the multivariate controller for multiple-input, multiple-output (MIMO) manufacturing processes. Castillo et al. (2002) proposed a multivariate double EWMA controller for drifting processes. Moreover, Tseng et al. (2013) proposed a multivariate EWMA controller for a linear dynamic process. However, both of these methods require the process model to be known, which is not the case in many applications. Additionally, they are mainly designed for multivariate time series and their extension to images is not trivial. Liu et al. (2019) proposed an image-based control method for additive manufacturing. This method first extracts some engineering features (e.g., contrast, energy, etc.) from the images, and then builds the process model connecting the control variables with the extracted features. After the process model is estimated, a PID controller is applied to control the process. This approach, however, may have two drawbacks: First, extraction of engineering features requires domain knowledge that may not be available in all applications; and second, the engineering features may not capture all essential information of images, which may lead to an inaccurate process model, and consequently, to poor control performance. Therefore, there is a need for a control framework for HD data that can address these issues.

To design an effective control strategy for an HD response, we first need to estimate

the functional relationship between control variables and the HD response sequence, offline, and then, obtain the control law by minimizing a proper control objective function, online. The main challenges in achieving this two-step control strategy are 1) the high-dimensionality of the model coefficients and the response variable, which may lead to overfitting; 2) the Spatio-temporal structure of the response sequence, and 3) the non-iid noises in the system.

Recently, multilinear algebra has been used to address the high-dimensionality challenge in regression modeling by exploiting the embedded low-dimensional (LD) structure of HD data. Zhou et al. (2013) employed PARAFAC/CANDECOMP (CP) decomposition to estimate a tensor regression model between a scalar output and HD inputs. The CP decomposition uses the sum of rank-1 tensors to approximate a tensor (Kiers et al., 2000). Li et al., (2018) applied a more general tensor decomposition, known as Tucker decomposition (Kolda et al., 2009) in building the regression model. To predict tensor outputs, Lock et al. (2018) developed a tensor-on-tensor regression approach using CP decomposition with an adaptive approach for a learning basis. That is, instead of fixing the input and output span basis when learning the regression coefficients, their method adaptively learns the input and output span basis from data. With this adaptive learning basic technique, the tensor-on-tensor regression method can handle the case when there is no prior knowledge of the input and output data, which significantly increases the flexibility and adaptability of the method. However, their approach only handles single tensor input and cannot be easily extended to multiple tensor inputs. To address this issue, Reisi et al. (2019) proposed multiple tensor-on-tensor regression that can efficiently handle multiple input tensors. However, this approach cannot be used for regression modeling in

control systems because of two reasons: First, it inherently assumes that input tensors are independent, which is not the case in control systems where the current response is a function of past responses as well as the control variables. Second, their approach ignores the Spatio-temporal structure of the error term that is important when dealing with an image sequence. Yan et al. (2019) took the Spatio-temporal structure of the error term into consideration by assuming that it follows a tensor normal distribution and estimated the parameters using the one-step tensor decomposition regression (OTDR) approach. However, this approach can only handle single scalar input and cannot be generalized to multiple tensor inputs.

The overarching goal of this chapter is to propose a methodology for designing and deploying an optimal control strategy that can handle an HD output and both LD and HD control variables. In the offline estimation step, we develop a novel tensor-based regression/time-series modeling technique to address the foregoing challenges. Borrowing the idea from the autoregressive-moving average with exogenous terms (ARMAX) model (Hannan et al. 1970), we assume the current output frame (tensor) is linearly correlated with several past output frames (tensors) and the immediate past control variables. To avoid estimating a large number of parameters and hence, the overfitting issue, we assume each input coefficient can be represented by an LD core tensor, input span basis matrices, and output span basis matrices. We develop efficient algorithms for estimating the core tensors and learning all LD basis matrices. For the online control part, we use an optimization model with a squared loss to obtain the optimal control law.

The rest of the chapter is organized as follows: In Section 2, we formulate the tensor-based ARMAX model and illustrate the detailed procedure for parameter learning. Then,

based on the trained model, we develop the HD control strategy for online control. In Section 3, using simulations, we validate the proposed methodology and compare it with several benchmarks both in terms of offline estimation and online control. A case study on online control of the overlay errors in semiconductor manufacturing is presented in Section 4. Finally, the chapter is concluded in Section 5.

3.2 Methodology

The proposed image-based control framework consists of two steps: the off-line estimation step to establish the relationship between the tensor response sequence and control variables, followed by the online control step that determines the optimal control law. In both steps, we consider a general setting, in which both response and control variables are in the tensor form. Each step is elaborated in the following subsections.

3.2.1 Offline estimation of relation function

Assume a set of training data of size $m + p$ is available, which includes a sequence of response data, denoted by the tensor $\mathcal{Y}_t \in R^{Q_1 \times \ldots \times Q_d} (t = 1, \ldots, m + p)$ and a sequence of control variables, denoted by $\mathcal{X}_t \in R^{P_1 \times \ldots \times P_s} (t = 1, \ldots m + p)$ collected over time. To consider both the temporal information of $\mathcal{Y}_t$, and its relationship with $\mathcal{X}_t$, we develop a tensor-based ARMAX (*p*, *q*, *l*) time-series model, where *p*, *q*, *l* are orders of the ARMAX model. In particular, p represents the order of the AR part, q denotes the order of the MA part, and l represents the order of the input data. The ARMAX (*p*, *q*, *l*) model defines the relationship between the current response tensor $\mathcal{Y}_t$, the past p response tensor observations, i.e., $\mathcal{Y}_{t-j}, j = 1, \ldots, p$, as well as the control variable $\mathcal{X}_t$, using the linear form given by

$$\mathcal{Y}_t = \Sigma_{j=1}^{p} \mathcal{Y}_{t-j} * \mathcal{A}_j + \Sigma_{n=0}^{l-1} \mathcal{X}_{t-n} * \mathcal{B}_n + \delta E_t, \quad (1)$$

where E_t represents the tensor of random noises, $\mathcal{A}_j \in R^{Q_1 \times \ldots \times Q_d \times Q_1 \times \ldots \times Q_d}$ $(j = 1, \ldots, p)$ and $\mathcal{B}_n \in R^{P_1 \times \ldots \times P_s \times Q_1 \times \ldots \times Q_d}$ $(n = 0, \ldots, l-1)$ are the coefficients for the corresponding input, the operator * is the contraction product of two tensors defined as (Reisi et al., 2019):

$$(\mathcal{X}_{t-n} * \mathcal{B}_n)_{q_1 \ldots q_d} = \Sigma_{p_1,\ldots,p_l} \mathcal{X}_{t-n,p_1,\ldots,p_l} \mathcal{B}_{n,p_1,\ldots,p_l,q_1,\ldots,q_d}.$$

To achieve a more compact representation, we can combine the *m* tensor observations over time into higher-order tensors denoted by $\tilde{\mathcal{Y}}_{(-j)} \in R^{Q_1 \times \ldots \times Q_d \times m}; j = 1, \ldots, p$, $\tilde{\mathcal{X}}_{(-j)} \in R^{P_1 \times \ldots \times P_l \times m}; j = 1, \ldots, p$, and $\mathcal{E} \in R^{Q_1 \times \ldots \times Q_d \times M}$. For example, $\tilde{\mathcal{Y}}_{(-j)}$ includes images $\{\mathcal{Y}_{p+1-j}, \ldots, \mathcal{Y}_{m-j}\}$ and $\tilde{\mathcal{Y}}_{(0)}$ is the set of images $\{\mathcal{Y}_{p+1}, \ldots, \mathcal{Y}_m\}$. Then, the ARMAX (*p*, *q*, *l*) model can be rewritten as

$$\tilde{\mathcal{Y}}_0 = \Sigma_{j=1}^{p} \tilde{\mathcal{Y}}_{-j} * \mathcal{A}_j + \Sigma_{n=0}^{l-1} \tilde{\mathcal{X}}_{-n} * \mathcal{B}_n + \delta \mathcal{E}\,. \quad (2)$$

To model the Spatio-temporal structure of the noise, it is assumed that $\mathcal{E}$ follow a tensor normal distribution as $\mathcal{E} \sim N(0, \Sigma_1, \Sigma_2, \ldots, \Sigma_{d+1})$, or equivalently $e = vec(\mathcal{E}) \sim N(0, \Sigma_{d+1} \otimes \ldots \otimes \Sigma_1)$, where $\Sigma_1, \ldots, \Sigma_d$ represent the spatial correlation of the noise that is defined by the following kernel function $\Sigma_{k|i_1,i_2} = \exp\left(-\theta \left\| r_{i_1} - r_{i_2} \right\|^2\right), k = 1, \ldots, d$, with $(r_{i_1} - r_{i_2})$ representing the distance between points i_1 and i_2. Σ_{d+1} captures the temporal (between-sample) variation that can be estimated from data. The tensor coefficients can be estimated by minimizing the negative likelihood function. However, since the dimensions of $\mathcal{A}_j; j \in \{1, \ldots, p\}$ and $\mathcal{B}_n; n \in \{0, \ldots, l-1\}$ are too high, estimating such a large number of parameters results in severe overfitting and is often

intractable. In reality, due to the structured correlation between the inputs and the response, we can assume all these coefficients lie in an LD space and can be expanded using a set of basis matrices via a tensor product. That is, it is assumed that both coefficients can be expanded by

$$\mathcal{B}_n = \mathcal{C}_{B_n} \times_1 \dots U_{B_n l} \times_{l+1} V_{B_n 1} \times_{l+2} \dots \times_{l+d} V_{B_n d}, \quad (3)$$

$$\mathcal{A}_j = \mathcal{C}_j \times_1 U_{j1} \times_2 \dots U_{jd} \times_{d+1} V_{j1} \times_{d+2} \dots \times_{2d} V_{jd}, \quad (4)$$

where $\mathcal{C}_{B_n} \in R^{\tilde{P}_1 \times \dots \times \tilde{P}_s \times \tilde{Q}_1 \times \dots \times \tilde{Q}_d}$ and $\mathcal{C}_j \in R^{\tilde{Q}_1 \times \dots \times \tilde{Q}_d \times \tilde{Q}_1 \times \dots \times \tilde{Q}_d}$ are core tensors with $\tilde{Q}_i \ll Q_i$.; $U = \{U_{ji}: j = 1, \dots, p; i = 1, \dots, d, U_{B_n i}: i = 1, \dots, s, n = 1, \dots, l\}$ is a set of basis matrices that spans the j^{th} input space; and $V = \{V_{ji}, i = 1, \dots d\}$ is a set of basis matrices that spans the output space. Thus, the estimation of HD coefficients $\mathcal{A}_j$ and $\mathcal{B}_n$ is transformed into learning the corresponding core tensors $\mathcal{C}$ and the basis matrices in the sets U and V. In this chapter, we allow U_{ji} to be learned directly from the input spaces and $U_{B_n i}$ to be truncated identity matrices since the control variables are usually independent of each other. Unlike the multiple tensor-on-tensor frameworks proposed in Reisi et al., (2019), our input data, $\tilde{\mathcal{Y}}_{-j}$, are highly serially correlated. If we treat these input data independently, the rank deficiency of the input will make the problem ill-conditioned. Therefore, we set $U_{1i} = U_{2i} = \dots = U_{pi} = U_i \; ; i = 1, \dots, d$. Note that even if we set the U bases to be the same for all input tensors, learning the core tensor $\mathcal{C}_j$, and the basis matrices in V provide sufficient degrees of freedom to learn the HD coefficients.

The basis matrices U_j can be learned using Tucker decomposition (Hitchcock et al., 1927) through the following optimization model:

$$\{\mathcal{D}_j, U_1, \dots, U_d\} = argmin_{D_j,\{U_j\}} \left\| \bar{\mathcal{Y}} - \mathcal{D}_j \times_1 U_1 \dots \times_d U_d \right\|_F^2, \quad (5)$$

where $\bar{\mathcal{Y}}$ is the mean of all response tensors over all different time stamps. After obtaining U_js, our next task is to estimate the core tensors C_j $(j = 1, \dots, p)$, C_{B_n} $(n = 0, \dots, l-1)$, and the basis matrices $V_{ji} (j = 1, \dots, p, i = 1, \dots, d)$, and $V_{B_n i} (n = 0, \dots, l-1, i = 1, \dots, d)$.

Inspired by the work of Yan et al., (2019), we can learn the core tensors and the output span basis matrices, i.e., $\mathcal{C}_j, \mathcal{C}_B, \{V_{ji}\}$, simultaneously. Moreover, instead of the orthogonality constraint on output span basis matrices, we apply the weighted constraint given by $V_{ji}^T \Sigma_i^{-1} V_{ji} = I$ and $V_{Bi}^T \Sigma_i^{-1} V_{Bi} = I$. These constraints ensure the closed-form solution in each iteration and guarantee a similar spatial covariance structure for the estimated basis matrices. Given U_js, the following likelihood function can be used to estimate the remaining parameters:

$$\begin{aligned} \underset{\mathcal{C}_j, \mathcal{C}_B, \{V_{ji}\}}{\operatorname{argmin}} \Bigg\{ \Bigg(Y_0 - \sum_{j=1}^{p} \left(Z_j \otimes V_{jd} \dots V_{j1}\right) vec\left(\mathcal{C}_j\right) \\ - \sum_{n=0}^{l-1} \left(X_{-n} \otimes V_{B_n d} \dots V_{B_n 1}\right) vec\left(\mathcal{C}_{B_n}\right) \Bigg)^T \left(\Sigma_{d+1} \otimes \dots \Sigma_1\right)^{-1} \Bigg(Y_0 \\ - \sum_{j=1}^{p} \left(Z_j \otimes V_{jd} \dots V_{j1}\right) vec\left(\mathcal{C}_j\right) - \sum_{n=0}^{l-1} \left(X_{-n} \otimes V_{B_n d} \dots V_{B_n 1}\right) vec\left(\mathcal{C}_{B_n}\right) \Bigg) \Bigg\} \end{aligned} \quad (6)$$

$$s.t. V_{ji}^T \Sigma_i^{-1} V_{ji} = I \text{ and } V_{Bi}^T \Sigma_i^{-1} V_{Bi} = I,$$

where Y_0 and Y_{-j} denote the transpose of the d-mode unfold of $\tilde{\mathcal{Y}}_0$ and $\tilde{\mathcal{Y}}_{-j}$, respectively, and X_{-n} denotes the transpose of the l-mode unfolding of $\tilde{\mathcal{X}}_{-n}$. The mode-j matricization of tensor $\mathcal{R} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_j \times \ldots \times I_n}$ is $\boldsymbol{R}_{(j)} \in \mathbb{R}^{I_j \times I_{-j}}$, where $I_{-j} = I_1 \times I_2 \times \ldots \times I_{j-1} \times I_{j+1} \times \ldots \times I_n$. Moreover, we define $Z_j := \left(U_{jd} \otimes \ldots \otimes U_{j1}\right)^T Y_{-j}$.

To efficiently optimize (6), we combine the alternating least square (ALS) with the block coordinate descent (BCD) methods (ALS-BCD). The details of the optimization algorithm for learning the core tensor as well as basis matrices are given in Propositions 1 and 2.

Proposition 1. *Given* V_{ji} $(j = 1, \ldots, d, i = 1, \ldots, d)$, V_{Bi} $(i = 1, \ldots, d)$, Z_j $(j = 1, \ldots, p)$ *and* Σ_k $(k = 1, \ldots, d+1)$ *a reshaped form of the core tensors* $\mathcal{C}_j$ *and* $\mathcal{C}_B$ *can be estimated by solving Problem (6), and the solutions are given by*

$$\mathcal{C}_j = R_j \times_1 V_{j1}^T \Sigma_1^{-1} \times_2 \ldots \times_d V_{jd}^T \Sigma_d^{-1} \left(Z_j^T \Sigma_{d+1}^{-1} Z_j\right)^{-1} Z_j^T \Sigma_{d+1}^{-1} \tag{7}$$

$$\mathcal{C}_B = R_B \times_1 V_{i1}^T \Sigma_1^{-1} \times_2 \ldots \times_d V_{id}^T \Sigma_d^{-1} (X_0^T \Sigma_{d+1}^{-1} X_0)^{-1} X_0^T \Sigma_{d+1}^{-1} \tag{8}$$

where $R_j = \tilde{\mathcal{Y}}_0 - \Sigma_{i \neq j}^{p} \tilde{\mathcal{Y}}_{-i} * \mathcal{A}_i - \Sigma_{n=0}^{l-1} \tilde{\mathcal{X}}_{-n} * \mathcal{B}_n$ *and* $R_{B_n} = \tilde{\mathcal{Y}}_0 - \Sigma_{i=1}^{p} \tilde{\mathcal{Y}}_{-j} * \mathcal{A}_j - \Sigma_{p \neq n}^{l-1} \tilde{\mathcal{X}}_{-p} * \mathcal{B}_p$.

The proof of this proposition is given in Appendix B.1.

From Proposition 1, we know that if the output span basis matrices are given, the core tensor can be learned from (7) and (8). To optimize V_{ji}, we use the following proposition.

Proposition 2. *Given the core tensor* $\mathcal{C}$*, basis matrix* U*, and* V_{ki} $(k \neq j)$*,*

a) we can minimize the negative log-likelihood function in (6) *by maximizing the projected scores norm in* (9) *and* (10)*,*

$$\text{argmax}_{\{V_{jk}\}} \left\| R_j \times_1 V_{j1}^T \Sigma_1^{-1} \dots V_{jd}^T \Sigma_d^{-1} \times_{d+1} X_{j,d+1} \right\|; \tag{9}$$

$$s.t. V_{jk}^T \Sigma_k^{-1} V_{jk} = I,$$

$$\text{argmax}_{\{V_{jk}\}} \left\| R_B \times_1 V_{B1}^T \Sigma_1^{-1} \dots V_{Bd}^T \Sigma_d^{-1} \times_{d+1} X_{B,d+1} \right\|; \tag{10}$$

$$s.t. V_{Bk}^T \Sigma_k^{-1} V_{Bk} = I,$$

where $X_{j,d+1}$ *and* $X_{B,d+1}$ *are computed by using the Cholesky decomposition, that is,*

$$X_{j,d+1} X_{j,d+1}^T = \Sigma_{d+1}^{-1} Z_j \left(Z_j^T \Sigma_{d+1}^{-1} Z_j \right)^{-1} Z_j^T \Sigma_{d+1}^{-1}, \tag{11}$$

$$X_{B_n,d+1} X_{B_n,d+1}^T = \Sigma_{d+1}^{-1} X_{-n} (X_{-n}^T \Sigma_{d+1}^{-1} X_{-n})^{-1} X_{-n}^T \Sigma_{d+1}^{-1}. \tag{12}$$

b) the maximizers of (9) *and* (10) *are* $V_{jk} = \Sigma_k^{\frac{1}{2}} \tilde{V}_{jk}$*, and* $V_{Bk} = \Sigma_k^{\frac{1}{2}} \tilde{V}_{Bk}$*, respectively, where* $\tilde{V}_{jk}$ *and* $\tilde{V}_{Bk}$ *are the first* P_k *eigenvectors of* $\Sigma_k^{-\frac{1}{2}} \boldsymbol{W}_{jk}$*, and* $\Sigma_k^{-\frac{1}{2}} \boldsymbol{W}_{Bk}$*.* $\boldsymbol{W}_{jk}$ *is the k mode unfolding of* $\mathcal{W}_{jk} := R_j \times_1 V_{j1}^T \Sigma_1^{-1} \dots \times_{k-1} V_{j,k-1}^T \Sigma_{k-1}^{-1} \times_{k+1} V_{j,k+1}^T \Sigma_{k+1}^{-1}$*, and* $\boldsymbol{W}_{Bk}$ *is the k mode unfolding of* $\mathcal{W}_{Bk} := R_B \times_1 V_{B1}^T \Sigma_1^{-1} \dots \times_{k-1} V_{B,k-1}^T \Sigma_{k-1}^{-1} \times_{k+1} V_{B,k+1}^T \Sigma_{k+1}^{-1} \times_d V_{Bd}^T \Sigma_d^{-1} \times_{d+1} X_{B,d+1}$.

The simplified proof of this proposition is shown in Appendix B.2.

Note that even though the estimated parameters by using Proposition 1 and 2 are not necessarily unique because of the identifiability issue in tensor regression, all different estimations tend to estimate the same mean value for the output. Therefore, as the main purpose of the offline estimation step is to predict the future output, the lack of identifiability would not be problematic. When the uniqueness of estimated parameters is important, we should add more constraints such as the sparsity of the core tensor. A more detailed discussion on this can be found in Anandkumar et al. (2015).

Using Propositions 1 and 2, our proposed estimation procedure for the tensor-based ARMAX model is summarized in Algorithm 1.

Algorithm 1. The estimation procedure for the tensor-based ARMAX model

Initialize C_j , C_B, V_{ji}, V_{Bi} for all i, j
Estimate U_{ij} using Tucker decomposition as shown in (5)
Calculate $Z_j = \left(U_{jd} \otimes \ldots \otimes U_{j1}\right)^T Y_{-j}$
Compute Σ_k^{-1} , $\Sigma_k^{1/2}$, $\Sigma_k^{-1/2}$ and compute X_{jd+1} , X_{Bd+1} by Cholesky decomposition as shown in (11) and (12).
Compute $\mathcal{A}_j$ and $\mathcal{B}$ for all j and set a_0 equals to the objective function in (7)
do
    Estimate V_{ji} for all i, j iteratively using Proposition 2.
    Estimate C_j , C_B using Proposition 1.
    Compute $\mathcal{A}_j$ and $\mathcal{B}_i$ for all i, j and set a_k equals to the objective function in (6)
while $|a_{k+1} - a_k| > \epsilon$

Note that the objective function in (6) is always non-negative and for each BCD iteration, so that it always drives our objective function downhill in each iteration. Therefore, the ALS-BCD algorithm always converges to a stationary point. In order to choose the set of tuning parameters including the covariance parameter θ, the rank $\tilde{Q}$, and the order p and q, we use the Bayesian Information Criterion (BIC).

To estimate the between-sample covariance matrix, we use the two-step regression approach summarized in Algorithm 2, which is a widely used approach in parameter estimation for the univariate ARMA model (Hannan et al. 1980).

Algorithm 2. The estimation procedure for the between-sample covariance matrix

1: Assume Σ_{d+1} is the identity matrix, build the initial using identity Σ_{d+1} model

2: Calculate $u_t = \mathcal{Y}_t - \Sigma_{j=1}^{p} \mathcal{Y}_{t-j} * \mathcal{A}_j - \Sigma_{n=0}^{l-1} \mathcal{X}_{t-n} * \mathcal{B}_n$

3: Vectorize all the residual tensors and partition them into different groups. In particular, group i include the tensor $\left[u_i, u_{q+i}, \dots, u_{\left[\frac{N}{q}\right]+i}\right]$. Here, N is the sample size, q is the order in ARMA model and $\left[\frac{N}{q}\right]$ is the floor function i.e., the smallest integer less than $\frac{N}{q}$.

4: Calculate within-group and between-group covariance matrix by

$$S_{d,ii} = \Sigma_{j=1}^{\frac{N}{q}} \left(vec(u_{ij}) - vec(\bar{u}_i)\right)\left(vec(u_{ij}) - vec(\bar{u}_i)\right)^T$$

$$S_{d,ik} = \Sigma_{j=1}^{\frac{N}{q}} \left(vec(u_{ij}) - vec(\bar{u}_i)\right)\left(vec(u_{kj}) - vec(\bar{u}_k)\right)^T$$

5: Calculate $a_{ij} = \min_{a_{ij}}(\|S_{d,ij} - a_{ij}(\Sigma_1 \otimes \Sigma_2 \dots \otimes \Sigma_d)\|)$ and set $\Sigma_{q,d+1}(i,j) = a_{ij}(i \in [1, \dots, q], j \in [1, \dots q])$

6: Form the super diagonal matrix Σ_{d+1} by computing the Kronecker product of $\Sigma_{q,d+1}$ with an appropriate identity matrix.

7: Use new Σ_{d+1} build the model again

8: Repeat 2-7 until convergence

Here, N denotes the total number of training data and S_d matrix is the intermediate resulting matrix. The intuition behind this algorithm is that we first estimate the empirical autocovariance matrix (for vectorized elements). Then, as the spatial covariance matrix estimates are fixed, we tune the elements in the between-sample covariance matrix (i.e., a_{ij}) to best approximate the empirical auto covariance matrix. Then, using the Kronecker product, each element in the between-sample covariance matrix will become a sub matrix in the overall autocovariance matrix by multiplying it with an identity matrix.

3.2.2 Online control

Once the tensor-based ARMAX model is estimated in the offline estimation step, it can be used for prediction and control. The goal of the optimal control is to find a control law that minimizes the expected difference between the response and the target value. For our proposed tensor-based model, the control objective function of the one-step-ahead optimal control is given by

$$J(\mathcal{X}_t) = \min_{\mathcal{X}_t} E\left\|\hat{\mathcal{Y}}_{t+1}(\mathcal{X}_t) - T\right\|_F^2, \tag{13}$$

where, $\hat{\mathcal{Y}}_{t+1}$ is the one-step-ahead predicted tensor using the estimated ARMAX model, T is the target tensor, and E(·) is the expectation operator. Based on this control objective function, the control law can be obtained by using proposition 3.

Proposition 3. *Minimizing the mean square error loss function in (13) is equivalent to solve the equality in* $E\mathcal{Y}_{t+1}(\mathcal{X}_t) = T$*, and therefore the optimal control action can be expressed as*

$$\begin{aligned} vec(\mathcal{X}_t) &= (U_{Bl} \otimes \ldots \otimes U_{B1}) C_B^{-1} (V_{Bd} \otimes \ldots V_{B2} \otimes V_{B1})^T vec(R_{Bt}) \\ &= C_B^{-1} (V_{Bd} \otimes \ldots V_{B2} \otimes V_{B1})^T vec(R_{Bt}), \end{aligned} \tag{14}$$

where the $C_B \in R^{\tilde{P} \times \tilde{Q}}$ *is the unfolded core tensor* $\mathcal{C}_B$ *with* $\tilde{P} = \Pi_{j=1}^{l} \tilde{P}_j$ *and* $\tilde{Q} = \Pi_{j=1}^{d} \tilde{Q}_j$ *and* $R_{Bt} = T - \Sigma_{j=1}^{p} \mathcal{Y}_{t+1-j} * \mathcal{A}_j$.

The proof of proposition 3 is shown in Appendix B.3.

3.2.3 Controllability discussion

The controllability analysis is essential because it reflects whether the target output can be achieved in finite time by adjusting the control variables. To analyze the controllability of the proposed control scheme, we exploit the LD subspaces that can effectively represent both the response and control variables using Proposition 4. We derive the controllability condition for our image-based control using the existing condition of the controllability for the state-space model. In particular, we try to reformate our image-based controller to the state-space form and then directly use the state-space model's conclusion.

Proposition 4. *Let* $\tilde{A} = \begin{bmatrix} (U_d^T V_{1d} \otimes \dots \otimes U_1^T V_{11})\tilde{C}_{\mathcal{A}_1} \dots & (U_d^T V_{1d} \otimes \dots \otimes U_1^T V_{11})\tilde{C}_{\mathcal{A}_p} \\ \boldsymbol{I} & \mathbf{0} \end{bmatrix}$ *and* $\tilde{B} = \begin{bmatrix} \left(U_d^T V_{Bd} \otimes \dots \otimes U_1^T V_{B_1 1}\right)\tilde{C}_{\mathcal{B}_1} \dots \left(U_d^T V_{B_2 d} \otimes \dots \otimes U_1^T V_{B_2 1}\right)\tilde{C}_{\mathcal{B}_l} & 0 \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$, *where* $\tilde{C}_{\mathcal{A}_1}, \dots, \tilde{C}_{\mathcal{A}_p}$ *and* $\tilde{C}_{\mathcal{B}_1}, \dots, \tilde{C}_{\mathcal{B}_l}$ *are metricized core tensors. Without loss of generality, we assume* $p > l$*. Then the system is controllable using the proposed control scheme in (14) if and only if* $\left[\tilde{B}\ \tilde{A}\tilde{B}\ \tilde{A}^2\tilde{B}\ \dots \tilde{A}^{pq_1 \dots q_d - 1}\tilde{B}\right]$ *has full-row rank.*

The proof of proposition 4 is shown in Appendix B.4.

3.3 Performance evaluation using simulations

In this section, we conduct simulations to evaluate the performance of the proposed image-based control method. We perform this study in two steps: first, we evaluate the performance of the proposed offline estimation approach, and then, we study the overall performance of the proposed control scheme including the offline estimation and online control. For the first study, we choose the state-of-art MTOT (Reisi et al., 2019) as a

benchmark to compare with our estimation method. For the overall methodology, we compare our method with three different methods: the univariate controller (Hannan et al., 1980) designated by "UVC", the image-based control using PCA features (Chen et al., 1998) designated by "PCAC", and the image-based control using engineered features (Liu et al., 2019) designated by "EFC". The univariate controller uses the average of all the pixels to transform the sequence of images to a univariate time-series used to derive the control law. In the PCA-based control benchmark, first, all images are vectorized, and using PCA the first few principal component scores are extracted as features. Then, these features are used to build a multivariate control model. In the last benchmark, engineering features such as contrast, and energy are extracted and used to derive the control law.

Following Yan et al. (2019), in our simulation study, we consider two response types, namely, the wave-shaped surface and point cloud of truncated cones. For each case, we generate a sequence of training data with length N_{tr} and a sequence of test data with length N_{te} as described in section 3.3.1. Using the training sequence, we estimate the coefficients of the ARMAX model by applying our proposed estimation method. In Section 3.3.2, the test sequence is used to evaluate the performance of our estimation method, and in Section 3.3.3, the test sequence is used to evaluate the online control method.

3.3.1 Data generation

Case 1. Wave-shape surface simulation. We assume $l = 1$, and $p = 2$ in the ARMAX model and simulate a sequence of 2D responses denoted by the matrix $\mathbf{Y}_t = \left[y_t\left(\frac{i_1}{I_1}, \frac{i_2}{I_2}\right)\right]$; $i_1 = 1, \dots, I_1; i_2 = 1, \dots, I_2,$ with $I_1 = 100,\ I_2 = 50$. The responses are generated according to the following linear model: $y_t = \mathcal{A}_1 * y_{t-1} + \mathcal{A}_2 * y_{t-2} +$

$(\mathcal{C}_B \times_1 \boldsymbol{V}^1 \times_2 \boldsymbol{V}^2 \times_3 \boldsymbol{X}_t) + \mathcal{E}_t$, where $\mathcal{A}_1 = \mathcal{C}_1 \times_1 \boldsymbol{V}^1 \times_2 \boldsymbol{V}^2 \times_3 \boldsymbol{V}^1 \times_4 \boldsymbol{V}^2$ and $\mathcal{A}_2 = \mathcal{C}_2 \times_1 \boldsymbol{V}^1 \times_2 \boldsymbol{V}^2 \times_3 \boldsymbol{V}^1 \times_4 \boldsymbol{V}^2$. In this simulation, we select three basis matrices, namely $\boldsymbol{V}^{(k)} = [\boldsymbol{v}_1^k, \boldsymbol{v}_2^k, \boldsymbol{v}_3^k]$ with $\boldsymbol{v}_\alpha^{(k)} = \left[\sin\left(\frac{\pi\alpha}{n}\right), \sin\left(\frac{2\pi\alpha}{n}\right), \dots, \sin\left(\frac{n\pi\alpha}{n}\right)\right]^T, \alpha = 1,2,3$. The two mode-3 slices of $\mathcal{C}_B \in R^{3*3*2}$ are randomly generated from a normal distribution, $N(0,0.3)$. The 2D input matrices $\boldsymbol{X}_t \in R^{2\times1}, t = p, \dots, N_{tr}$, or N_{te} are randomly sampled from the standard normal distribution $N(0,1)$. We generate the noises from the tensor normal distribution defined by $e = vec(\mathcal{E}) \sim N(0, \Sigma_{d+1} \otimes \dots \otimes \Sigma_1)$, where the spatial correlation structure on the covariance matrix is given by $\Sigma_1 = \Sigma_2 = \cdots = \Sigma_d = \exp(-\theta\|r_{i1} - r_{i2}\|^2)$ and the temporal correlation is defined based on the between sample covariance matrix, Σ_{d+1}. This covariance matrix is the autocovariance matrix of the MA(1) model with the MA coefficient equal to $(1-\mu) = 0.7$. We use two settings to generate noises: 1) strong spatial correlation ($\theta = 10^{-4}$) with the MA error of $E_t = \epsilon_t - (1-\mu)\epsilon_{t-1}$, $\epsilon_t \sim \sigma_\epsilon N(0, \Sigma_d \otimes \dots \otimes \Sigma_1)$, and $\mu = 0.3$; and 2) weak spatial correlation ($\theta = 10^4$) with the MA error of $E_t = \epsilon_t - (1-\mu)\epsilon_{t-1}$, $\epsilon_t \sim \sigma_\epsilon N(0, \Sigma_d \otimes \dots \otimes \Sigma_1)$, and $\mu = 0.3$. We define the Signal-to-Noise Ratio (SNR) value as $\frac{\Sigma_{i=1}^{N_{tr}} \|Y_{tr,i}\|_F^2}{\Sigma_{i=1}^{N_{tr}} \|E_i\|_F^2}$. Therefore, the SNR values for strong and weak spatial correlations are $1.09 * 10^{-9}$ and $6.04 * 10^{-6}$, respectively. We generate $N_{tr} = 200$ samples as training data and $N_{te} = 200$ samples as testing data, according to the foregoing procedure.

Case 2. Truncated cylinder point cloud simulation. We simulate a sequence of truncated cylinder point clouds in a 3D cylindrical coordinate system (r, ϕ, z), where $\phi \in [0,2\pi]$, and $z \in [0,1]$. The corresponding r values at $(\phi, z) = \left(\frac{2\pi i_1}{I_1}, \frac{i_2}{I_2}\right), i_1 = 1, \dots, I_1; i_2 =$

$1, \dots, I_2$ with $I_1 = 100, I_2 = 50$ for the t^{th} sample is recorded as the response. We simulate the mean patterns of the point cloud surface $\boldsymbol{P}_t$ such that $r(\phi, z) = 1$ for any pair of (ϕ, z). Then, we add the variational pattern generated by the tensor time-series sequence. Specifically, the following model is used to generate the tensor sequence:

$$\boldsymbol{P}_t = \mathcal{A}_1 * \boldsymbol{P}_{t-1} + \mathcal{A}_2 * \boldsymbol{P}_{t-2} + \left(\mathcal{C}_B \times_1 \boldsymbol{V}^1 \times_2 \boldsymbol{V}^2 \times_3 \widetilde{\boldsymbol{X}}_t\right) + \left(\mathcal{C}_B \times_1 \boldsymbol{V}^1 \times_2 \boldsymbol{V}^2 \times_3 \widetilde{\boldsymbol{X}}_0\right) + \mathcal{E}_t,$$

where, $\boldsymbol{P}_t \in R^{I_1 \times I_2}$ represents the variational pattern at time t, and $\widetilde{\boldsymbol{X}}_t \in R^{4\times 1}$ is a control vector. The four mode-3 slices of $\mathcal{C}_B \in R^{3*3*4}$ are randomly generated from a normal distribution, $N(0,0.3)$. Similarly, to generate $\mathcal{A}_1, \mathcal{A}_2, \boldsymbol{V}^1, \boldsymbol{V}^2$, and $\mathcal{E}_t$, we follow the procedure described in Case 1. We generate $N_{tr} = 200$ samples as training data and $N_{te} = 200$ samples as test data.

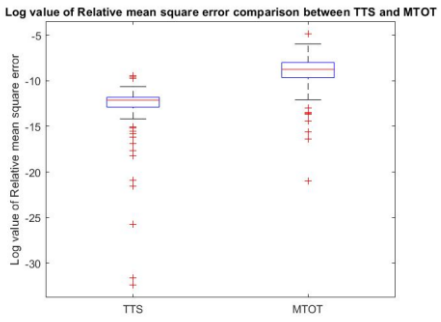
3.3.2 Simulation study for offline estimation

In this study, our goal is to recover the coefficients $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{B} = \mathcal{C}_B \times_1 \boldsymbol{V}^1 \times_2 \boldsymbol{V}^2$ from the training sequence. As discussed earlier, we consider two scenarios with two different noise settings. We apply our method denoted by tensor time-series (TTS) as well as the benchmark, MTOT, on the generated training sequences and compute the relative mean squared error (RMSE), defined by $\frac{\|\widehat{Y_{te}} - Y_{te}\|_F^2}{\|Y_{te}\|_F^2}$ of each method using the test sequences. The average RMSE values across 100 replications are reported in Table 4.

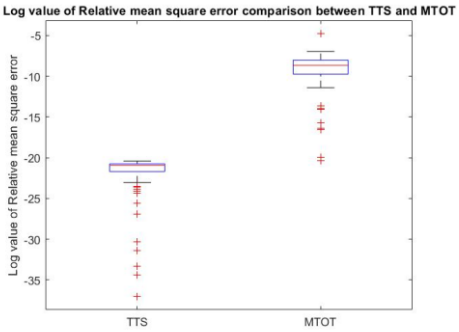
Table 4. Comparison between our proposed method and MTOT

	Case 1				Case 2			
	Setting 1		Setting 2		Setting 1		Setting 2	
$1/SNR$	$6.04 * 10^{-6}$		$1.09 * 10^{-9}$		$1.99 * 10^{-6}$		$3.52 * 10^{-10}$	
Method	MTOT	TTS	MTOT	TTS	MTOT	TTS	MTOT	TTS
Average RMSE	3.5×10^{-4}	$\mathbf{8.2 \times 10^{-6}}$	3.14×10^{-4}	$\mathbf{6.9 \times 10^{-10}}$	2.4×10^{-4}	$\mathbf{6.9 \times 10^{-6}}$	2.1×10^{-4}	$\mathbf{5.4 \times 10^{-7}}$
Computation time	$534.88s$	$\mathbf{13.26s}$	$536.84s$	$\mathbf{12.24s}$	$978s$	$\mathbf{15.14s}$	$1128s$	$\mathbf{25.28s}$

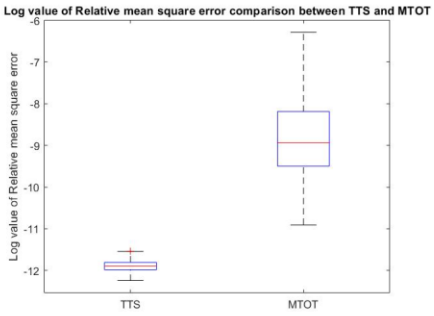
From Table 4, we found that when either the temporal or Spatio-temporal correlation structure of the error is high, our method will outperform the MTOT method. Additionally, the boxplots of the logarithm of RMSE values are shown in Figure 8.



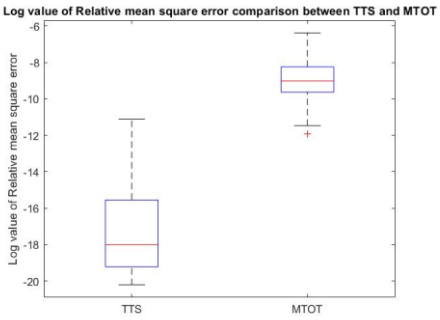
Case 1. Setting 1



Case 1. Setting 2



Case 2. Setting 1



Case 2. Setting 2

Figure 8. Boxplot of log RMSE comparison between TTS and MTOT

From Table 4 and the boxplots in Figure 8, it is clear that our proposed tensor ARMAX method outperforms MTOT under both cases and setting combinations. The minimum improvement obtained from our method is in the order magnitude of 10. This is because, unlike MTOT, our method takes both the spatial and temporal structures of error terms into consideration. Furthermore, Table 4 shows that the estimation time of the proposed method is significantly less than that of the MTOT indicating that our method converges in much fewer iterations than MTOT. This is because our method utilizes the correlation information in the time-series noise.

3.3.3 Simulation study for evaluating overall performance.

In this section, we use training and test sequences generated in the previous section to evaluate the overall performance of the proposed tensor-based control (TSC) methodology including both the offline estimation and the online control components. For each setting, we first learn the unknown model coefficients from the training sequence. Then, based on the estimated coefficients, we perform one-step-ahead control on the test sequence with the target output 0. We use the UVC, PCAC, EFC methods as benchmarks to compare their performance with TSC in terms of the steady-state relative mean square deviation from target defined as RMSD = $\frac{\Sigma_{i=N_{te}/2}^{N_{te}} \|y_{te,ac}(i)-T\|_F^2}{\Sigma_{i=N_{te}/2}^{N_{te}} \|y_{te}(i)-T\|_F^2}$. The average RMSD are reported in Table 5.

Table 5. Average relative mean squared deviation from target

	Case 1		Case 2	
RMD	Setting1	Setting2	Setting1	Setting2
Image-based controller	0.23	0.21	1.0361	0.67
Univariate controller	5.83×10^{7}	3.65×10^{11}	65.05	2.47×10^{5}
PCA-based method	1.94×10^{4}	3.76×10^{11}	37.31	1.13
Feature-based method	7.2×10^{19}	2.90×10^{27}	2.89×10^{31}	4.32×10^{35}

Additionally, to compare the variability of the methods, we plot the boxplots of RMD values in Figure 9.

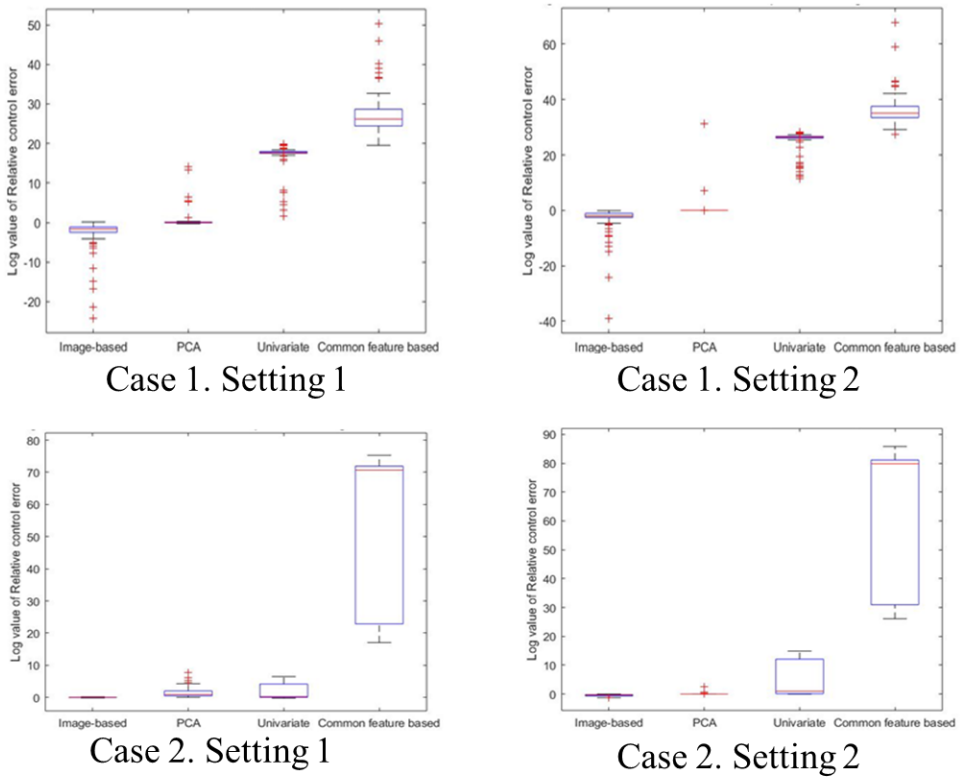


Figure 9. Boxplot of RMD values

In Figure 10, the blue line represents the RMSD from the target without any control strategy and the red line represents the RMSD from the target with applying our proposed methodology. From this, it is clear that our proposed method can significantly improve the overall control performance.

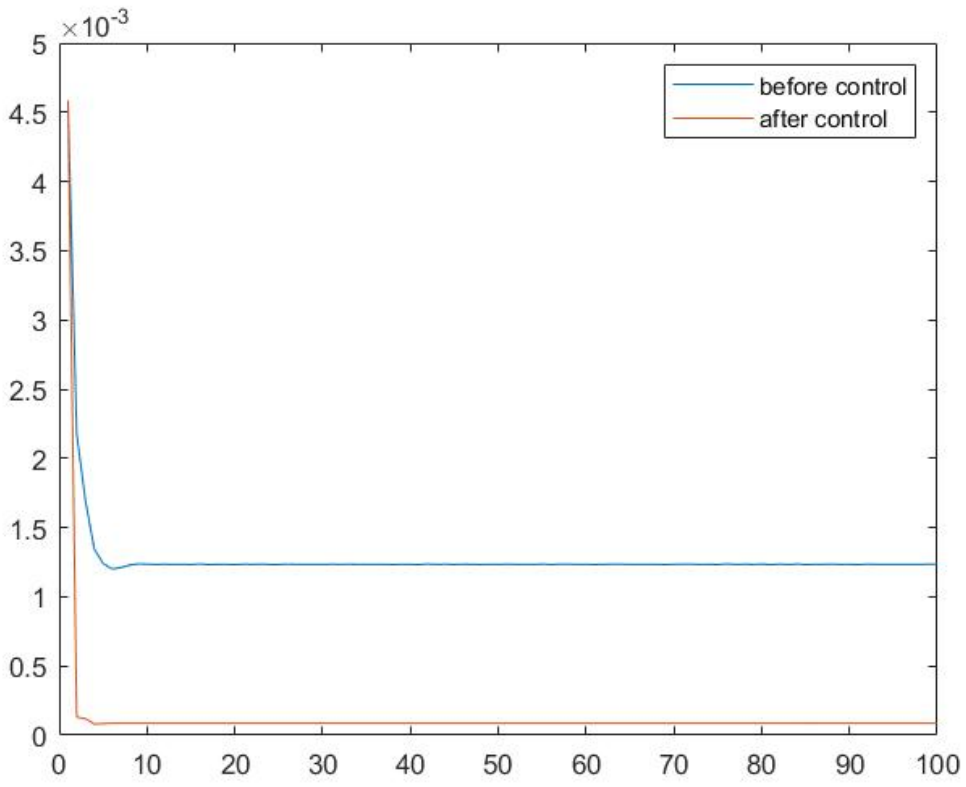
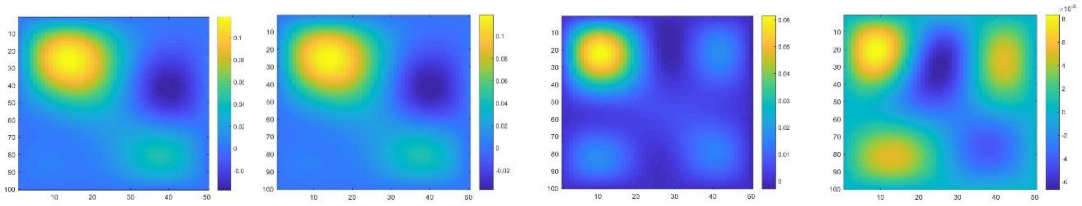
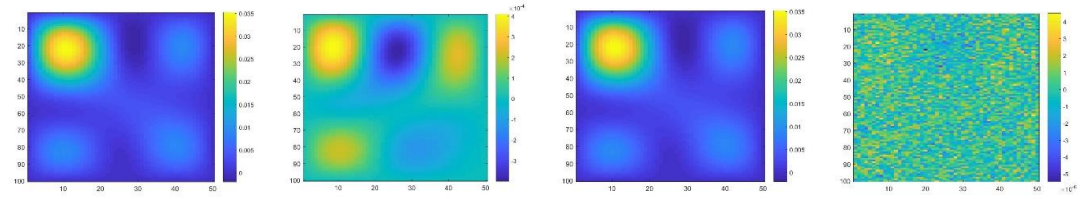


Figure 10. RMSD from the target of Case 2, Setting 2

From the above table, our method will perform much better than the rest controller, which demonstrates the effectiveness of our control method. To visualize the effectiveness of the control method, we plot the snapshots of a sample image sequence before and after control in Figure 11.



(a) T=1 before control (b) T=1 after control (c) T=10 before control (d) T=10 after control



(e) T=15 before control (f) T=15 after control (g) T=20 before control (h) T=20 after control

Figure 11. Comparison between before and after control results

From the figure, it can be seen that as time progresses the control procedure drives the image into the target, i.e., zero, until the spatial correlation is removed, and it reaches the steady-state of random noises. This can be better illustrated by a video clip given in the online appendix.

3.4 Case study

The photolithography process is a critical stage in semiconductor manufacturing and silicon wafer production. The main objective of the photolithography process is to carve the designed circuit pattern onto the wafer surface. During the lithography process, with the help of the optical system, the patterns on a mask will be projected onto a thin layer of photoresist material on the wafer. The photoresist material will quickly solidify when exposed to light. Then, the unexposed material is washed away. The entire wafer is comprised of m identical rectangular fields, on which the light exposure is performed in each layer. After completing one layer, the procedure is repeated to print the subsequent

layers.

One of the most critical quality measurements in the lithography process is the overlay error, which represents the misalignment between the photoresist materials in two subsequent layers. Overlay error at each measurement location can be represented by a 2D vector, which denotes the relative locational difference between two adjacent layers with the start point on the previous layer and the end point on the current layer. To fully characterize the misalignment across the wafer, the overlay measurements are often taken at multiple sites within every field as shown in Figure 12. In this figure, the grids represent the boundaries of the fields; the vectors are the 2D overlay vectors, whose projection on each axis denotes the overlay error on the corresponding axis. Figure 13 shows the wafer coordinate system used for error decomposition.

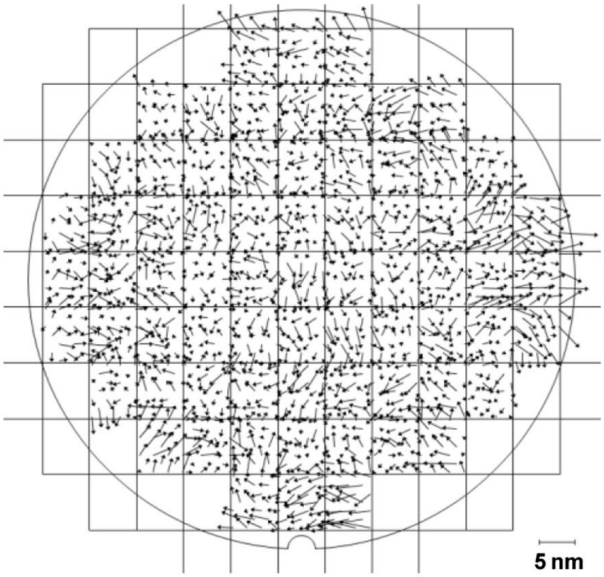


Figure 12. Illustration of the overlay measurements on a wafer (Brunner et al. 2013)

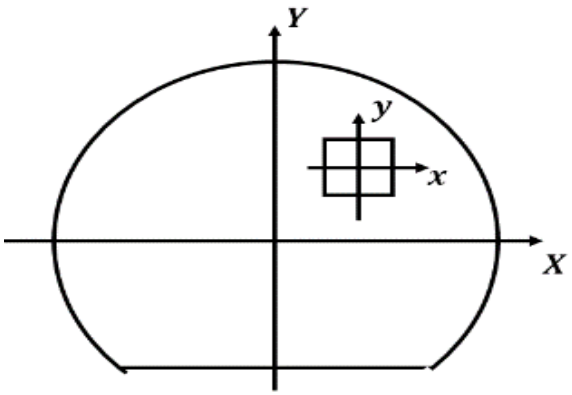


Figure 13. Illustration of the wafer coordinate system and the field coordinate system

The overlay measurements can be represented as $N \times 2$ image data, where N is the total number of overlay measurements on each wafer. The control variables include the critical settings of the lithography process including the wafer position, lens height that are adjusted by a set of setting knobs on the machine. In this case study, overlay data are generated from a simulator endorsed by a well-known semiconductor company. The detailed procedure of generating data is illustrated in Appendix B.5.

Using this simulator, we generate one training sequence with 3000 samples, one test sequence with 200 samples for validation of offline estimation, and one test sequence with 200 samples for validation of online control. We first estimate the model using the training sequence. From the experience of the subject matter expert, we set $p = q = l = 1$, and model between sample variation (i.e., the error term), by an IMA process. The optimal rank and Σ_3 can be estimated using the proposed BIC criteria in Section 2.3. We apply our proposed method to learn the coefficients $\mathcal{A}_1$ and $\mathcal{B}$ from the training sequence. After obtaining $\mathcal{A}_1$ and $\mathcal{B}$, we use the validation data to compute the relative mean square test error and compare it with the MTOT, as shown in Table 6.

Table 6 Relative test MSE of estimated models

	MTOT	TTS
$\frac{\left\|\widehat{Y_{te}} - Y_{te}\right\|_F^2}{\left\|Y_{te}\right\|_F^2}$	6.47×10^{-5}	7.19×10^{-11}

As expected, the results from Table 6 show that our proposed TTS significantly outperforms the MTOT method since it is not designed for time-series data, which is validated by Figure 14.

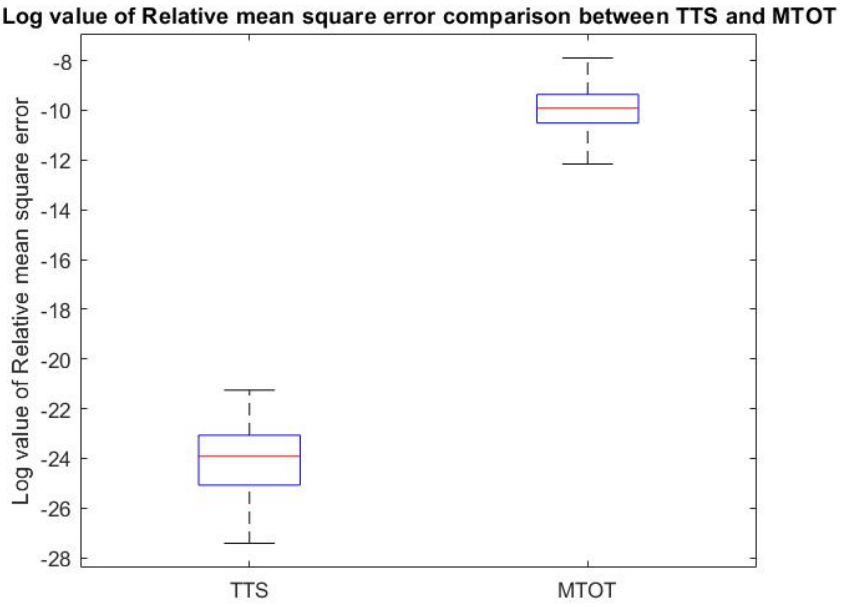


Figure 14. Boxplot of log RMSE comparison between TTS and MTOT

After the model is validated, we perform online control on the test sequence and record relative mean squared deviation (RMSD) from the target, shown in Table 7.

Table 7. Resulting RMSD by applying different control methods

	Image-based control	Univariate control	PCA-Based method
$\frac{\left\|Y_{te,\,after\_control} - T\right\|_F^2}{\left\|Y_{te,before\_control} - T\right\|_F^2}$	6.63×10^{-5}	1.71	1.70

The target value in the online control model for the overlay error is set to 0. The boxplots of RMSD values are also plotted in Figure 15.

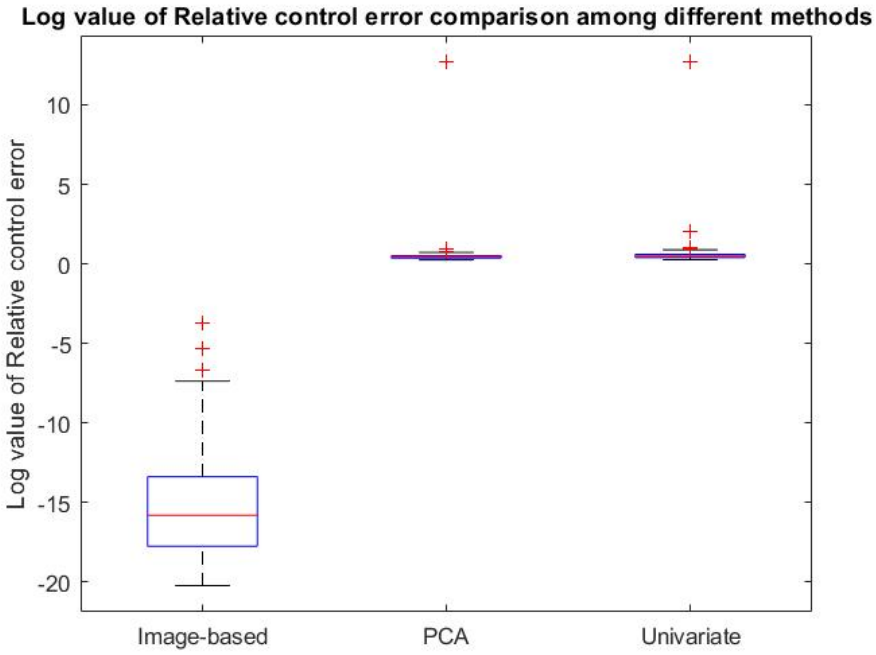


Figure 15. Boxplot of log RMSE comparison between TTS and MTOT

From the results reported in Table 7 and Figure 15, it is clear that our method outperforms the benchmarks including UVC and PCAC. This is because, in the offline model estimation, our method can extract more representative features that capture Spatio-temporal information of both the control and response variables, resulting in a more accurate control model. Additionally, to study the performance of the proposed control model over time, we plot the sequence of the log RMSD values for the test data in Figure 16.

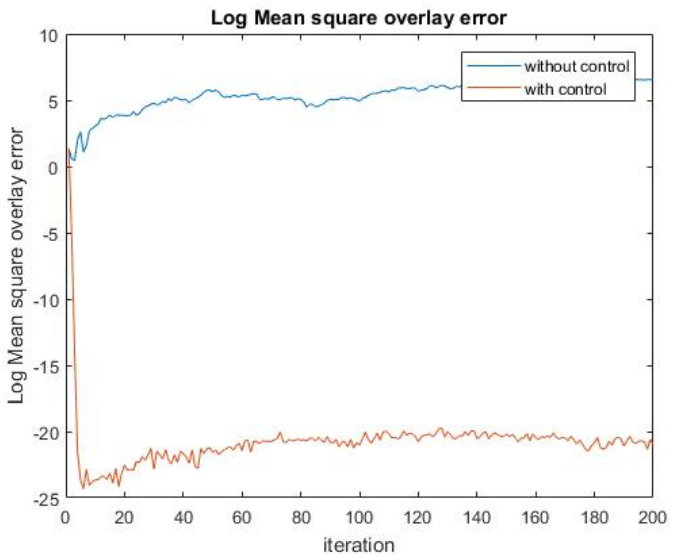


Figure 16. Log RMSD of overlay error over time

In this figure, the blue and red lines represent the log RMSD of the overlay error with and without applying the control strategy. It is clear from the figure that the proposed control method can significantly keep the overlay error close to zero that is the desired target.

3.5 Conclusion

This chapter proposed a systematic approach for feedback control when the response, as well as the control variables, are in the form of HD data. Our proposed method consists of two steps, the off-line estimation step that focused on establishing the link model between the output and control input variables using a training dataset. This was followed by the online control step providing the optimal control law to ensure the minimal deviation of the response from its target. In the offline estimation step, a novel tensor time-series modeling approach was proposed. After obtaining the model coefficients, an optimization model was used to minimize the control objective function.

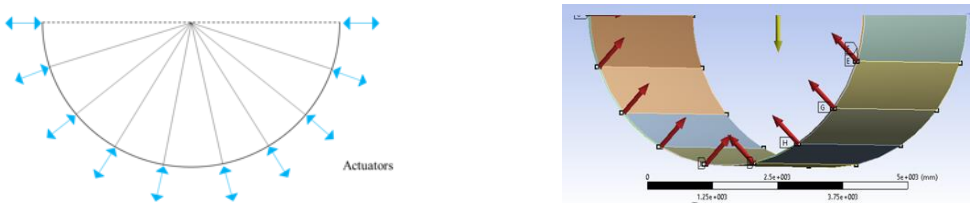
To validate the effectiveness of the proposed method, we conducted simulations as well as a case study in semiconductor manufacturing. In the simulation study, both the offline estimation and control actions obtained by our method outperformed the benchmark methods. We also applied our tensor-based control to a set of surrogate data generated from an overlay simulator in semiconductor manufacturing. The result showed that our proposed method can tremendously decrease the overall overlay error, which implies that our proposed method is effective in designing and deploying control systems with HD data. Our proposed control framework was developed based on the premise that the HD response (e.g., image) is spatially smooth. More research is required to study the development of control strategies for HD data with non-smooth characteristics, such as textured images.

CHAPTER 4. FEA MODEL BASED CAUTIOUS AUTOMATIC OPTIMAL SHAPE CONTROL FOR FUSELAGE ASSEMBLY

4.1 Introduction

Composite parts have been widely used in the aerospace industry (Gates, 2007) due to their superior performance and unique characteristics, such as high stiffness-to-weight ratio, low life-cycle cost, and potentially longer life. In an aircraft assembly, multiple composite parts need to be assembled with ultra-high precision. However, since the composite parts are fabricated by multiple suppliers in multiple batches, there always have some natural dimensional variabilities in each composite part (Gates, 2007). To achieve ultra-high precision assembly, we need to develop effective methodologies for optimal shape adjustment of these composite parts to compensate for their dimensional variations and initial deviation.

A half to half fuselage assembly is one of the critical tasks in an aircraft assembly process. To achieve automatic optimal shape control, a number of actuators are used to push or pull the fuselage to compensate for its shape distortions (Wen et al., 2018), (Yue et al. 2018). Figure 17(a) shows a potential actuator placement strategy, while Figure 17(b) visualizes the forces exerted by these actuators.



(a) Potential actuator placement strategy (b) Schematic drawing of shape adjustment

Figure 17. Illustration for the fuselage shape control

In current industrial practice, actuators are employed to push and pull a fuselage for its shape adjustment. The magnitude of adjustments of those actuators is conducted by a trial-and-error method based on in-situ measurements of the fuselage. There are three limitations for the current practice: (i) low efficiency: a longer time and multiple trials may be needed to adjust those actuators to achieve a desired shape during the assembly process; (ii) non-optimality: the location and force of each actuator are non-optimal; (iii) highly-skilled engineers required: the skills of engineers will determine the quality and efficiency of an assembly process. The uncertainty of engineers' skills increases the uncertainties of the time and quality of the fuselage assembly.

Currently, numerous efforts have been conducted for modeling and analysis of dimensional variation and control for the full fuselage assembly. Wen et al. (Wen et al. 2018) first built a finite element analysis (FEA) model to mimic the physical properties of a real fuselage. Their model has been validated with the real experimental data using a full fuselage in an aircraft assembly plant. On top of this, they proposed a surrogate model based shape control method that considers uncertainties for full fuselage assembly (Yue et al., 2018). Furthermore, Du et al. (2019) proposed a sparse-learning method to automatically select the actuator locations. The main limitations of these existing methods (Yue et al., 2018) (Du et al., 2019) are that they require a complicated surrogate model which links the relationships among fuselage deviation, actuator locations, and forces of actuators. However, a surrogate model is time-consuming to train since the training data is collected from many FEA replications. Moreover, when the potential number of actuator locations increases, the required number of FEA replications will also increase. Therefore, a surrogate model can only be built on a limited number of prespecified and fixed potential

locations for the actuators, which limits the optimality and accuracy of the control performance. To address these issues, we propose an FEA model based automatic optimal shape control (AOSC) method for half fuselage assembly. In our proposed method, the model used to develop the optimal shape control is loaded from the FEA software directly, which is different from the existing methods (Yue et al., 2018) (Du et al., 2019) where a surrogate model is built to find the relationship between the shape deviation and the actuator adjustments. The proposed AOSC method has the following advantages: (i) The system equation is directly loaded from an FEA software, which can be obtained much faster and more efficiently than training a surrogate model from experimental data via FEA simulations. (ii) A surrogate model is an approximation of the FEA model; therefore, the system equation directly loaded from an FEA software has much higher accuracy than a surrogate model. (iii) The number of potential candidate locations of the actuators in our system equation is much larger than those in a surrogate model. (iv) A cautious control concept (Zhong et al., 2010) (Maciejowski et al., 2002) is used to address the model uncertainties due to the difference between the FEA model developed with a designed part and the real in-coming parts on the assembly floor. And finally, (v) The problem formulation to solve the optimal location and force of each actuator is convex, thus can be solved efficiently by CVX (Grant et al., 2008).

The remainder of this chapter is organized as follows. Section 2 provides a detailed illustration of the proposed AOSC framework. Then, Section 3 provides case studies to validate the performance of our proposed method. Finally, Section 4 concludes the chapter.

4.2 The Automatic Optimal Shape Control Framework

In this section, we first introduce our FEA based process model in section 4.2.1. Then, we elaborate on the proposed AOSC framework in section 4.2.2. Finally, we introduce how to address the model uncertainty with the cautious control concept in section 4.2.3.

4.2.1 FEA based process model

The linear elastic mechanical response of a fuselage is determined by the material property of the fuselage and the force applied to it. Their relationships can be described by the following global stiffness equation (Maciejowski et al., 2002)

$$\boldsymbol{KU} = \boldsymbol{F}_g + \boldsymbol{F}_r, \tag{1}$$

Let N be the total number of mesh nodes of a given fuselage in the FEA model. Then, $\boldsymbol{K} \in R^{6N\times 6N}$ is the global stiffness matrix; $\boldsymbol{U} = [\boldsymbol{u}_1; \dots; \boldsymbol{u}_N] \in R^{6N\times 1}$ is the nodal displacement vector assembled by aggerating the displacement vector $\boldsymbol{u}_i = \left[u_x^i, u_y^i, u_z^i, \omega_x^i, \omega_y^i, \omega_z^i\right]^T \in R^{6\times 1}, i \in \{1, .., N\}$ on each mesh node. The first three elements of $\boldsymbol{u}_i$ denote the three-dimensional linear displacements and the last three elements of $\boldsymbol{u}_i$ denote the three-dimensional angular displacements of the ith mesh node; $\boldsymbol{F}_g = \left[\boldsymbol{f}_{g1}; \dots; \boldsymbol{f}_{gN}\right] \in R^{6N\times 1}$ represents the gravity load, and $\boldsymbol{F}_r = [\boldsymbol{f}_{r1}; \dots; \boldsymbol{f}_{rN}] \in R^{6N\times 1}$ represents the load exerted by the fixture locating points and actuators. They are assembled in the same way as nodal displacement vector $\boldsymbol{U}$, by aggerating the gravity-induced load vector $\boldsymbol{f}_{gi} = \left[f_{gx}^i, f_{gy}^i, f_{gz}^i, \tau_{gx}^i, \tau_{gy}^i, \tau_{gz}^i\right]^T \in R^{6\times 1}, i \in \{1, .., N\}$ or the fixture

and actuator induced load vector $\boldsymbol{f}_{ri} = \left[f_{rx}^i, f_{ry}^i, f_{rz}^i, \tau_{rx}^i, \tau_{ry}^i, \tau_{rz}^i\right]^T \in R^{6\times 1}$, $i \in \{1, .., N\}$ on each mesh node. The first three elements of $\boldsymbol{f}_{gi}$, $\left[f_{gx}^i, f_{gy}^i, f_{gz}^i\right]^T$, denote the three-dimensional gravity-induced force; and the last three elements of $\boldsymbol{f}_{gi}$, $\left[\tau_{gx}^i, \tau_{gy}^i, \tau_{gz}^i\right]^T$, denote the three-dimensional gravity-induced torque of the ith mesh node. $\boldsymbol{f}_{ri}$ is defined similarly for its components as $\boldsymbol{f}_{gi}$. Notice that $\boldsymbol{f}_{ri} = \boldsymbol{0}$ if the ith mesh node is not used as a fixture locating point or an actuator location. We assume that fixtures restrict displacement vector $\boldsymbol{u}_i$ to be zero (if the ith mesh node is used as a fixture locating points) and the actuators can only apply forces instead of torques on the mesh nodes. In other words, the $\left[f_{rx}^i, f_{ry}^i, f_{rz}^i\right]^T$ can be non-zero only if there is an actuator or fixture locating point on the mesh node $i \in \{1, \dots, N\}$. The stiffness matrix ($\boldsymbol{K}$) and the gravity load ($\boldsymbol{F}_g$) are exported from the FEA simulation platform. Since the pre-specified fixture locating points restrict the corresponding linear displacement vectors to zero, a common practice in the FEA solution procedure (Grant et al., 2008) is to remove rows and columns corresponding to linear displacement vectors of fixture locating points in $\boldsymbol{K}$. We call this new stiffness matrix $\boldsymbol{K}^* \in R^{6N^*\times 6N^*}$, where $N^* = N - N_1$ and N_1 is the number of fixtures locating points. $\boldsymbol{K}^*$ is a positive definite matrix. Similarly, we remove the rows corresponding to linear displacement vectors of fixture locating points from original $\boldsymbol{U}$, $\boldsymbol{F}_g$, and $\boldsymbol{F}_r$ to obtain $\boldsymbol{U}^*$, $\boldsymbol{F}_g^*$, and $\boldsymbol{F}_r^*$, respectively. Then, Equation (1) becomes

$$\boldsymbol{K}^*\boldsymbol{U}^* = \boldsymbol{F}_g^* + \boldsymbol{F}_r^*, \tag{2}$$

Equation (2) will be used to design the automatic optimal shape control strategy.

4.2.2 FEA Model Based Automatic Optimal Shape Control Strategy

Our objective is to find $\boldsymbol{F}_r^*$ to achieve minimum total deviation δ^2, i.e.,

$$\delta^2 = (\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T)^T \boldsymbol{A} (\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T),$$

where $\boldsymbol{U}^* \in \boldsymbol{R}^{6N^*}$ represents the shape deviation induced by the gravity load and the load exerted by actuators, $\boldsymbol{U}_0^* \in \boldsymbol{R}^{6N^*}$ represents the initial shape deviation, and $\boldsymbol{U}_T \in \boldsymbol{R}^{6N^*}$ is the target shape. $\boldsymbol{A}$ is a diagonal matrix with only ones on locations of the linear displacement $\left[u_x^i, u_y^i, u_z^i\right]^T$ of mesh nodes that we are interested in, and all zeros otherwise, i.e.,

$$diag(\boldsymbol{A}) = \left[0,0,0, \dots, \underbrace{1,1,1,0,0,0}_{i} \dots, 0,0,0\right]^T$$

where, $diag(\boldsymbol{A})$ returns the diagonal elements of matrix $\boldsymbol{A}$.

In practice, the following physical constraints are required:

(i) First, $\boldsymbol{F}_r^*$ is a sparse vector with only nonzero elements in the actuator locations. Furthermore, assume that we only place actuators on the boundary of the fuselage, we have $\boldsymbol{F}_r^*(Non\ boundary\ point) = 0$.

(ii) Noting the fact that actuators only exert force but not torque on the fuselage. In other words, an actuator-induced torque vector on each mesh node i, which is $\boldsymbol{\tau}_i^* = \left[\tau_{rx}^i, \tau_{ry}^i, \tau_{rz}^i\right]^T$, should be zero. Thus, we have $\boldsymbol{\tau}_i^* = \boldsymbol{0}$, $i \in \{1, \dots, N^*\}$.

(iii) We further define the magnitude of the force vector on each node as $\|\boldsymbol{f}_i^*\| = \left\|\left[f_{rx}^i, f_{ry}^i, f_{rz}^i\right]^T\right\|_2$. Then the magnitude of the force vector for all mesh nodes is$[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T$. When we have n_a actuators, there are only n_a force vectors with nonzero elements. Thus, we have $\|[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T\|_0 = n_a$, where $\|\cdot\|_0$ is l_0 norm representing the number of nonzero entries in a vector.

(iv) In addition, the actuator exerted force should be bounded to avoid potential damages to the fuselage. This fact implies a constraint $\|\boldsymbol{f}_i^*\|_2 < F_{r_{UB}}, \forall i \in \{1, \dots, N^*\}$, where $F_{r_{UB}}$ are the upper bound of the actuator forces from engineering safety specifications.

Then, the optimal shape control problem can be formulated as

$$\min_{\boldsymbol{F}_r^*}(\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T)^T \boldsymbol{A}(\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T), \tag{3}$$

subject to:

$$\boldsymbol{K}^*\boldsymbol{U}^* = \boldsymbol{F}_g^* + \boldsymbol{F}_r^* \tag{3a}$$

$$\boldsymbol{F}_r^*(Non\ boundary\ point) = 0 \tag{3b}$$

$$\boldsymbol{\tau}_i^* = \boldsymbol{0}, i \in \{1, \dots, N^*\} \tag{3c}$$

$$\|\boldsymbol{f}_i^*\|_2 < F_{r_{UB}}, i \in \{1, \dots, N^*\} \tag{3d}$$

$$\boldsymbol{F}_r^* = [\boldsymbol{f}_{r1}^*; \dots; \boldsymbol{f}_{rN^*}^*] \tag{3e}$$

$$\boldsymbol{f}_{ri}^* = [\boldsymbol{f}_i^*; \boldsymbol{\tau}_i^*], \boldsymbol{f}_i^* = \left[f_{rx}^i, f_{ry}^i, f_{rz}^i\right]^T, \boldsymbol{\tau}_i^* = \left[\tau_{rx}^i, \tau_{ry}^i, \tau_{rz}^i\right]^T, i \in \{1, \dots, N^*\} \tag{3f}$$

$$\|[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T\|_0 = n_a. \tag{3g}$$

Unfortunately, the l_0 norm constraint $(3g)$ is generally non-convex, non-smooth, and NP-hard (Natarajan et al., 1995), which makes solving the optimization problem (3)

computationally intractable. Follow the similar procedure of (Du et al., 2019), we can transform the optimization problem (3) into a convex optimization problem with group lasso penalty:

$$\min_{\boldsymbol{F}_r^*}(\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T)^T \boldsymbol{A}(\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T) + \lambda \Sigma_{i=1}^{N^*} \|\boldsymbol{f}_i^*\|_2 \tag{4}$$

$$\text{subject to: Constraints } (3a) - (3f)$$

where λ is a tuning parameter, and its value needs to be selected to meet the constraint $\|[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T\|_0 = n_a$. If the tuning parameter λ is too large, we will have $\|[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T\|_0 < n_a$ and the control performance tends to become worse; when a small tuning parameter is chosen, we will have $\|[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T\|_0 > n_a$ and more than acceptable actuators will be selected. Algorithm 1 shows the AOSC algorithm.

Algorithm 1. The AOSC algorithm

(1) **Input:** parameters $\boldsymbol{A}, \boldsymbol{K}^*, \boldsymbol{U}_T^*, \boldsymbol{U}_0^*, \boldsymbol{F}_{\boldsymbol{g}}^*, F_{r_{UB}}, n_a$.

(2) **Initialize:** $\lambda_{min} = 0$, λ_{max} is set as a large enough arbitrary value

(3) Repeat

(4) $\lambda = (\lambda_{min} + \lambda_{max})/2$

(5) Calculate $\boldsymbol{F}_r^*$ by solving optimization problem (4) using the CVX software.

(6) If $\|[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T\|_0 = n_a$:

(7) Return λ and $\boldsymbol{F}_r^*$

(8) Else if $\|[\|\boldsymbol{f}_1^*\|_2, \dots, \|\boldsymbol{f}_{N^*}^*\|_2]^T\|_0 > n_a$:

(9) $\lambda_{min} = \lambda$

(10) Else

(11) $\lambda_{max} = \lambda$

(12) End

(13) **End**

According to Algorithm 1, we can obtain the optimal location and force of each actuator to achieve minimum total deviation using exactly n_a actuators.

The main difference between our proposed method and the current literature (Yue et al., 2018) (Du et al., 2019) (Haftka and Adelman, 1985) (Chee et al., 2002) (Burdisso and Haftka, 1990) (Hakim and Fuchs, 1996) (Burdisso and Haftka, 1989) (Ponslet et al., 1993) (Haftka and Adelman, 1985) is that our AOSC method is based on the FEA model. Since we can directly load the system equation from the FEA simulation platform, we do not need to build a surrogate model which has lower accuracy and lower efficiency. The schematic diagram of the proposed method is shown in Figure 18.

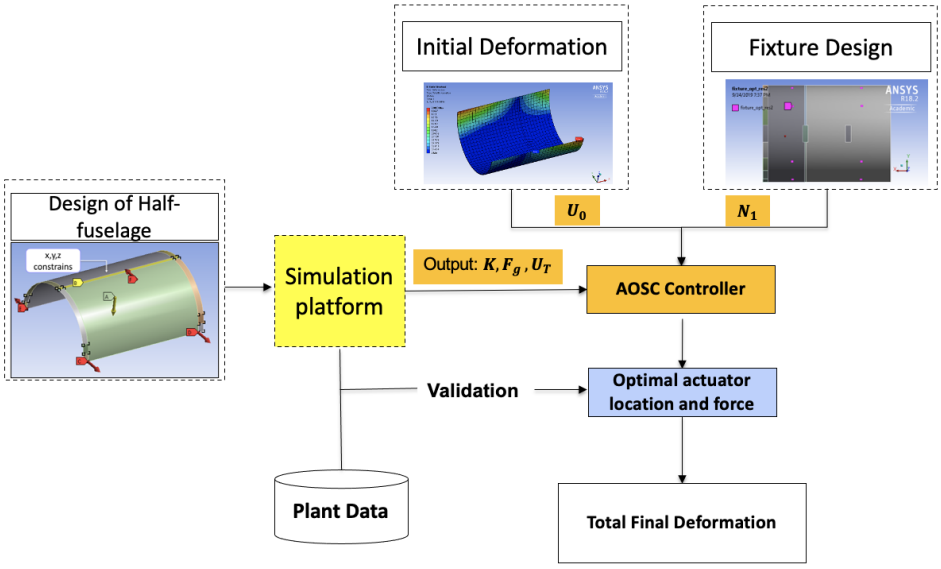


Figure 18. Schematic diagram of the proposed AOSC method

To apply the AOSC algorithm, we need to know several critical inputs in advance. First, we need to know the gravity-induced load and the stiffness matrix of the fuselage, which can be loaded from the FEA simulation platform. Also, the initial deviation of the fuselage needs to be specified. This information can be obtained from a 3D laser scan in a

production line. Moreover, the fixture locating points should also be specified according to the fixture design data. Based on these inputs, our AOSC algorithm will generate both the optimal actuator locations and their corresponding forces. We can use these actuators to push and pull the fuselage to obtain the optimal shape control results.

4.2.3 Problem formulation for Cautious AOSC.

In the previous section, we proposed a novel AOSC method to find each actuator location and calculate the force inserted by each actuator using the stiffness matrix obtained from the designed fuselage dimension and material properties. However, the real dimension and material properties of an in-coming fuselage may deviate from the engineering-designed data. Thus, a system model of the real fuselage may have some uncertainties in the stiffness matrix that is obtained from the FEA model.

To further improve the AOSC performance, we adopt the cautious control concept (Zhong et al., 2010) (Maciejowski et al., 2002). We assume that the stiffness matrix of incoming fuselages $\boldsymbol{K}^*$ is a random variable following a fixed distribution that can be estimated from the historical data. Then, the cautious control concept, which deals with the uncertainty by minimizing the expectation of the objective function, can be incorporated into our AOSC framework with the following formulation:

The solution of the optimal force will be a function of both the mean and standard deviations of the inversion of the stiffness matrix. In the cautious control strategy, we will first calculate the actuator locations from the original formulation (4). Then, we can calculate the force of each actuator by solving the following optimization problem:

$$\min_{\boldsymbol{F}_r^*} \mathbb{E}[(\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T)^T \boldsymbol{A}(\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T)] \tag{4'}$$

$$\text{subject to: Constraints } (3a) - (3f)$$

Since $\boldsymbol{K}^*$ is positive definite in constraint (3a), by plugging the constraint $\boldsymbol{U}^* = (\boldsymbol{K}^*)^{-1}\left(\boldsymbol{F}_g^* + \boldsymbol{F}_r^*\right)$ into the objective function $(4')$, we will have:

$$\min_{\boldsymbol{F}_r^*} \mathbb{E}\left[\left(\boldsymbol{F}_g^* + \boldsymbol{F}_r^*\right)^T (\boldsymbol{K}^*)^{-1}\boldsymbol{A}(\boldsymbol{K}^*)^{-1}\left(\boldsymbol{F}_g^* + \boldsymbol{F}_r^*\right) + 2(\boldsymbol{U}_0^* - \boldsymbol{U}_T)^T \boldsymbol{A}(\boldsymbol{K}^*)^{-1}\left(\boldsymbol{F}_g^* + \boldsymbol{F}_r^*\right)\right]$$

$$\text{subject to: Constraints } (3b) - (3f)$$

Since $\boldsymbol{K}^*$ is the only random variable, we have

$$\min_{\boldsymbol{F}_r^*}\left(\boldsymbol{F}_g^* + \boldsymbol{F}_r^*\right)^T \mathbb{E}[(\boldsymbol{K}^*)^{-1}\boldsymbol{A}(\boldsymbol{K}^*)^{-1}]\left(\boldsymbol{F}_g^* + \boldsymbol{F}_r^*\right) + 2(\boldsymbol{U}_0^* - \boldsymbol{U}_T)^T \boldsymbol{A}\mathbb{E}[(\boldsymbol{K}^*)^{-1}]\left(\boldsymbol{F}_g^* + \boldsymbol{F}_r^*\right) \tag{5}$$

$$\text{subject to: Constraints } (3b) - (3f)$$

where parameters $\mathbb{E}[(\boldsymbol{K}^*)^{-1}\boldsymbol{A}(\boldsymbol{K}^*)^{-1}]$ and $\mathbb{E}[(\boldsymbol{K}^*)^{-1}]$ can be calculated from the historical data. As can be seen from problem (5), instead of minimizing the quadratic loss function directly, we minimize the expectation of the quadratic loss function since the stiffness matrix is treated as a random matrix variable. This, on average, will ensure better performance than the non-cautious control strategy. Since problem (5) is convex, it can be solved efficiently by using convex optimization packages. Algorithm 2 shows the Cautious AOSC algorithm.

	Algorithm 2. The Cautious AOSC algorithm
(1)	**Input:** parameters $\boldsymbol{A}, \boldsymbol{U}_T^*, \boldsymbol{U}_0^*, \boldsymbol{F}_{\boldsymbol{g}}^*, F_{r_{UB}}$, $n_a, \mathbb{E}[(\boldsymbol{K}^*)^{-1}\boldsymbol{A}(\boldsymbol{K}^*)^{-1}]$ and $\mathbb{E}[(\boldsymbol{K}^*)^{-1}]$.
(2)	Calculate the AOSC optimal location of each actuator using Algorithm 1.
(3)	Calculate the optimal actuator forces $\boldsymbol{F}_r^*$ on those locations by solving optimization problem (5) using CVX software.

4.2.4 Discussion on the nonlinear efforts in the fuselage model

Notice that the FEA based process model can only describe the linear elastic mechanical response behavior of the fuselage. Thus, it is an interesting topic to discuss how the nonlinear effects may have an impact on the AOSC method.

In the FEA of structural mechanics, there are three major types of nonlinearity (Plumbridge et al., 2007):

(i) Geometric nonlinearity, which is due to large deformations, large strains.

(ii) Material nonlinearity, which is due to plasticity, creep, viscoplasticity/ viscoelasticity.

(iii) Boundary nonlinearity, which is due to the contact change.

It has been validated in Wen et al. (Chee et al., 2002) that the FEA model with linear material property and no boundary nonlinearity can mimic the physical properties of a real fuselage well. Therefore, we only need to consider the influence of geometric nonlinearity. In automatic control of the fuselage assembly process, there are strict limits of strain and stress allowed to be applied on the fuselage due to the high safety standards

(Chee et al., 2002) This guarantees the impact of geometric nonlinearity on the control performance to be negligible. The validation via empirical study is shown in section 3.3.

4.3 Case Study

In the case study, we build an FEA model of a half fuselage by adopting the same set of parameters used by Wen et al. (Wen et al., 2018), which was validated with real fuselage in practice. This FEA model serves as our ground truth to develop and validate the AOSC method. Under the small deformation assumption, the finite element model (Reddy, 2019) (Natarajan, 1995) for a composite part is shown in Equation (1). In Equation (1), the total stiffness matrix K and gravity-induced load F_g can be exported from the Ansys Workbench software. By using this FEA model, we will validate our proposed AOSC algorithm in section 4.3.1, and further demonstrate that the Cautious AOSC method outperforms the AOSC method when there exist uncertainties in the stiffness matrix K in section 4.3.2. The difference in control performance by using linear and nonlinear models is elaborated in section 4.3.3.

4.3.1 Case study results with the AOSC method.

In this section, five fuselages with different initial shape deviations are used to test the performance of the AOSC algorithm. The performance is measured in terms of maximum deviation among all nodes on the two edges of a fuselage that are shown in Figure 19.

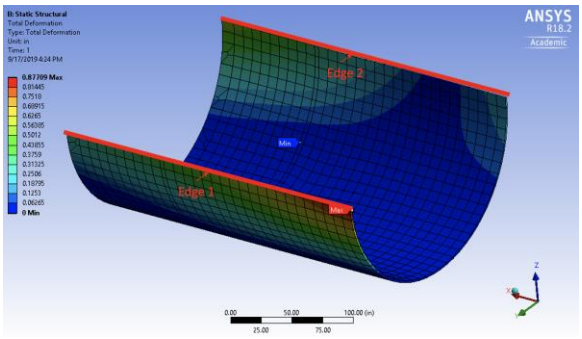


Figure 19. Two edges (in red color) of a half fuselage.

We first define the set of nodes on two edges as $E = \{i_1, \dots i_l\}$ where l is the total number of nodes on those two edges. The nodal linear displacement of nodes on those two edges can be extracted as:

$$\boldsymbol{U}_E^* = \boldsymbol{A}(\boldsymbol{U}^* + \boldsymbol{U}_0^* - \boldsymbol{U}_T),$$

where $\boldsymbol{A}$ is a diagonal matrix in which the mesh nodes on those two edges are 1 and all other nodes are 0. $\boldsymbol{U}_E^*$ has the following form:

$$\boldsymbol{U}_E^* = \left[u_x^{i_1}, u_y^{i_1}, u_z^{i_1}, \dots, u_x^{i_l}, u_y^{i_l}, u_z^{i_l}\right].$$

We further define the magnitude of the linear displacement vector on the ith node as $\|\boldsymbol{v}_i^*\| = \left\|\left[u_x^i, u_y^i, u_z^i\right]^T\right\|_2$. Then the magnitude of the linear displacement vector of nodes on those two edges is $\left[\left\|\boldsymbol{v}_{i_1}^*\right\|_2, \dots, \left\|\boldsymbol{v}_{i_l}^*\right\|_2\right]^T$. Finally, the maximum deviation can be defined as:

$$U_{Max}^* = \left\|\left[\left\|\boldsymbol{v}_{i_1}^*\right\|_2, \dots, \left\|\boldsymbol{v}_{i_l}^*\right\|_2\right]^T\right\|_\infty.$$

By using this definition, the U^*_{Max} reflects the maximum deviation among all mesh nodes interested. After applying these actuators on the in-coming half fuselage with different initial dimensions, we can obtain after-control maximum deviation of all points interested (MD-API) evaluated by linear and nonlinear models, respectively.

4.3.1.1 Control performance comparison between the industrial practice and the AOSC algorithm

In the current half fuselage assembly process, the industrial practice is that engineers use eight actuators to manually push and pull the fuselages to the target shape. Figure 20 shows the locations of the fixed fixture locating points and the locations of actuators based on the real fixture design and the actuators set up in the assembly station.

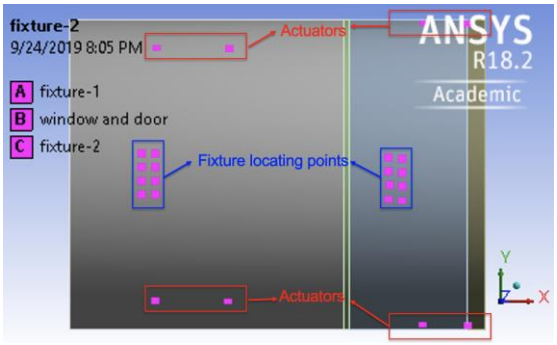


Figure 20. Fixed fixture and actuator locations in current industrial practice

To show the superiority of our algorithm, we conduct the following two comparisons on one incoming fuselage with the initial MD-API 14.45 inches:

(i) Comparison between control performance of current actuator locations and optimized actuator locations by the AOSC framework with the same number of actuators

In this comparison study, we will first use the current actuator locations and optimize the actuator forces to achieve its best achievable control performance, which

mimics the current industrial practice. Then, we adopt the same fixture locations and the same number of actuators but use the AOSC algorithm to find the optimal location and force of each actuator. The result is shown in Table 8. By using the same number of actuators (e.g. eight actuators in this study), the best performance of using current actuator locations is 1.99 inches, while our AOSC algorithm can reduce the MD-API to 0.068 inches.

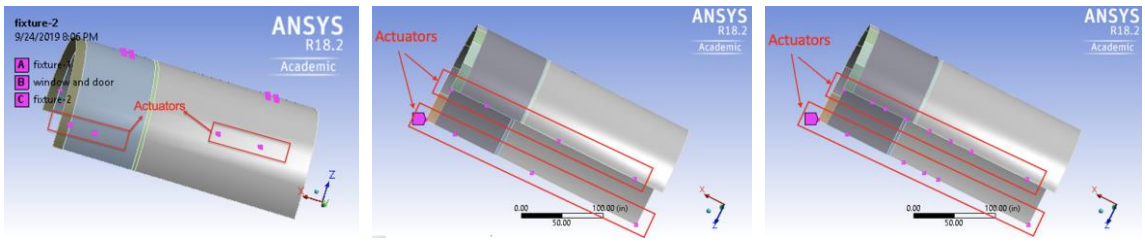
(ii) Comparison between control performance of optimized actuator locations by the AOSC framework with more actuators

In this study, we adopt the same fixture locations while further relax the number of actuators allowed in the AOSC algorithm to find the optimal location and force of each actuator. The results are shown in Table 8, which shows that we can further reduce the MD-API to 0.034 inches by using 14 actuators.

Table 8. Initial MD-API and control result comparison using industrial practice and AOSC algorithm

	Initial MD-API	Industrial practice	AOSC with 8 actuators	AOSC with 14 actuators
MD-API (inches)	14.45	1.99	0.068	0.034

The actuator locations of each case study above are shown in Figure 21



(a) Current location of 8 actuators (b) AOSC optimal location of 8 actuators (c) AOSC optimal location of 14 actuators

Figure 21. Current locations and AOSC optimal actuator locations

Figure 21 (a) shows the current actuator locations of 8 actuators in industrial practice, (b) shows the AOSC optimal location of 8 actuators and (c) shows the AOSC optimal location

of 14 actuators. The actuator locations given by AOSC are selected from the boundary and optimized by considering initial shape distortion. This makes full use of each actuator and thus leads to better control performance.

4.3.1.2 AOSC control performance evaluation

In this section, we will apply the AOSC algorithm on 5 incoming fuselages with different initial deviations. In the following discussions, the fixture locations are determined by the optimal fixture design algorithm in (Du et al. 2021). The comparison results between with control and without control are shown in Figure 22. In Figure 22, the x-axis represents the serial number of incoming fuselages, and the y-axis represents the MD-API of the fuselage. The result of with/without control is marked as blue/yellow respectively. From Figure 22, we can see that with reasonable amounts of actuators, the proposed AOSC algorithm can significantly reduce the MD-API from larger than 1 inch to smaller than 0.07 inches.

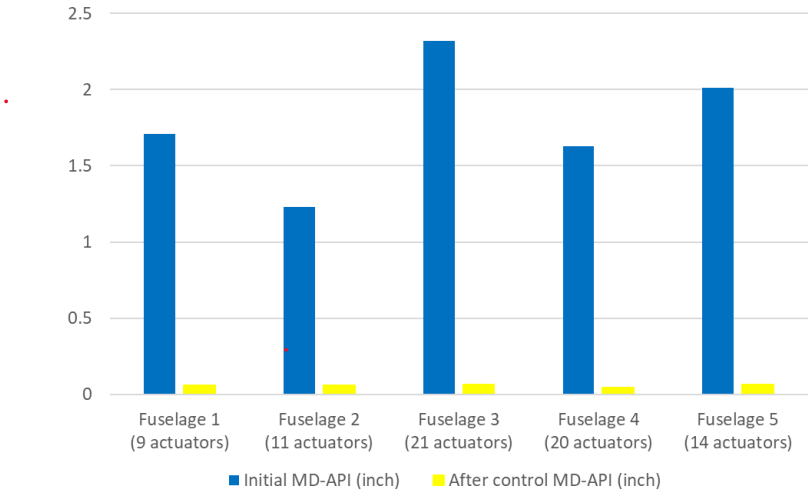
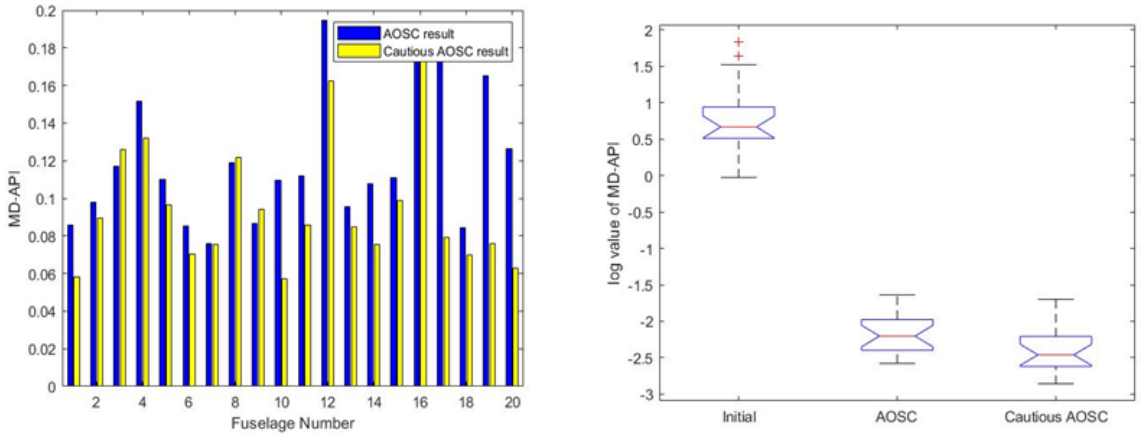


Figure 22. Control results for five incoming fuselages

4.3.2 Case study using Cautious AOSC.

In the AOSC algorithm, the stiffness matrix $\boldsymbol{K}$ is obtained from the designed fuselage. However, there always have some uncertainties in the stiffness matrix $\boldsymbol{K}$ in the real part due to the variation in initial shape deviation as well as the change of material properties of the fuselage. Thus, the stiffness matrix $\boldsymbol{K}$ will be different for each incoming fuselage. To address this challenge, we propose the cautious automatic optimal shape control (Cautious AOSC) algorithm. The effectiveness of this algorithm will be validated in this subsection.

In this study, we first load 20 stiffness matrices from 20 different fuselages with different dimensions. Then, we calculate $\mathbb{E}[(\boldsymbol{K}^*)^{-1}\boldsymbol{A}(\boldsymbol{K}^*)^{-1}]$ and $\mathbb{E}[(\boldsymbol{K}^*)^{-1}]$ from these 20 stiffness matrices and control these 20 fuselages by solving the optimization problem (5). The control results of these 20 fuselages are shown in Figure 7.



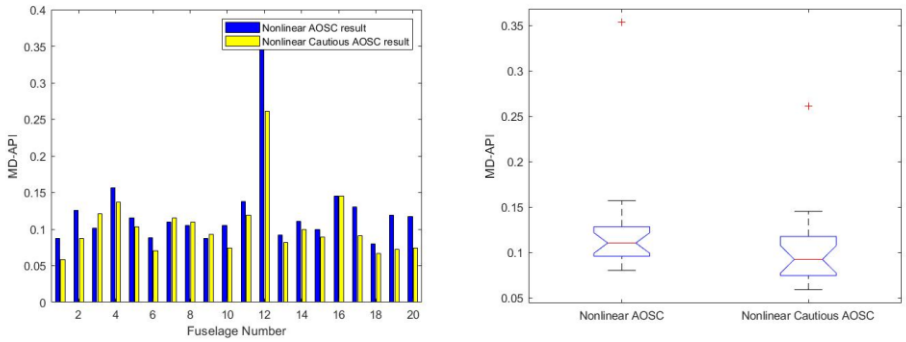
a) Bar Charts b) Box Plots

Figure 23. Control performance comparison between the Cautious AOSC and the AOSC algorithms

In Figure 23(a), the x-axis represents the serial number of fuselages, while the y-axis shows the MD-API of the fuselages. The blue bars represent the MD-API after applying the AOSC control strategy, while the yellow bars show the MD-API generated by the Cautious AOSC controller. The mean MD-API of these 20 fuselages for the AOSC algorithm is 0.1192 inches. However, by applying the Cautious AOSC algorithm, the mean MD-API can be reduced to 0.0950 inches and the improvement rate is 20.30%. From Figure 23(b), we find both the AOSC and the Cautious AOSC can significantly reduce the MD-API. Moreover, compared to the AOSC, the Cautious AOSC will generate better control results. This demonstrates that the proposed Cautious AOSC algorithm will provide better control results comparing with the AOSC algorithm.

4.3.3 Comparison between linear and nonlinear model

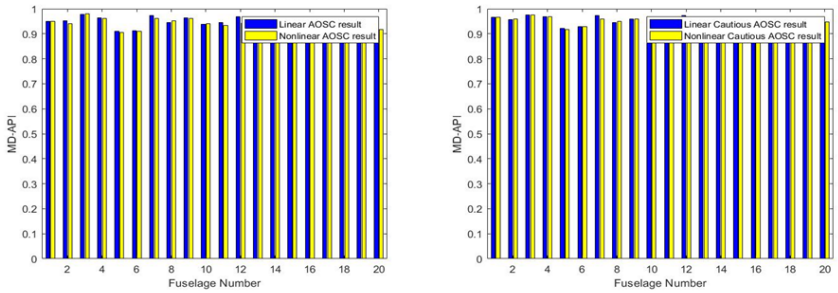
In this section, we show the nonlinear control results of the aforementioned 20 fuselages. The comparison result between linear and nonlinear models is shown in Figure 24. In Figure 24 (a), the nonlinear control results generated by the AOSC controller and the Cautious AOSC controller are shown by blue and yellow bars, respectively. Evaluated by the nonlinear model, the mean MD-API of the AOSC and the Cautious AOSC are 0.1234 inches and 0.1036 inches respectively. The improvement rate of Cautious AOSC over AOSC is 16.05%. This indicates the Cautious AOSC still outperforms the AOSC when evaluated by the nonlinear model.



(a) Bar charts (b) Box plots

Figure 24. Nonlinear control performance comparison between the Cautious AOSC and the AOSC algorithms

To quantitatively evaluate the difference between linear and nonlinear models, we define a new metric called improved MD-API, which is $\frac{\|U^*_{Max,in}-U^*_{Max,ac}\|_F^2}{\|U^*_{Max,in}\|_F^2} \times 100\%$. Here, $U^*_{Max,in}$ represents the MD-API of the initial incoming fuselage and $U^*_{Max,ac}$ represents the after-control MD-API. The improved MD-API of 20 fuselages by using the AOSC and the Cautious AOSC are shown in Figure 25 (a) and Figure 25(b). The results of using linear and nonlinear models are marked as blue/yellow respectively. From Figure 25, the difference between linear and nonlinear models is negligible.



(a) AOSC (b) Cautious AOSC

Figure 25 Control performance comparison between linear and nonlinear models

As shown in Table 9 , the mean improved MD-API of the AOSC controller is 88.98% for the linear model and 89.07% for the nonlinear model respectively, while the

mean improved MD-API of the Cautious AOSC controller is 91.56% for the linear model and 90.97% for the nonlinear model respectively.

Table 9. Comparison between the linear and nonlinear result of the AOSC and the Cautious AOSC controller

	AOSC controller	Cautious AOSC controller
Linear model result	88.98%	90.97%
Non-linear model result	89.07%	91.56%

This indicates that the linear model is a good approximation of the nonlinear model and our derivation of the optimization problem based on the linear model is reasonable.

4.4 Conclusion

This chapter proposes a novel AOSC system for the shape control of half fuselage assembly processes. In this system, the critical parameters of a new incoming fuselage will be exported from an FEA simulation platform. On top of these critical parameters, a convex optimization problem is formulated, which provides the optimal location and corresponding force for each actuator. Compared with the existing methods used in the fuselage assembly, our proposed AOSC algorithm is based on the FEA model, which is more efficient and optimal. Moreover, we further propose a cautions AOSC algorithm that takes model uncertainty into consideration. A set of case studies shows that our proposed algorithm outperforms the existing state-of-art methods.

APPENDIX A. Supplementary Materials for Chapter 2

A.1 Cautious control law derivation

To derive the cautious control law, we substitute the process model 5 into the loss function $J(u_t) = \mathbb{E}[(y_t - T)^2]$, and derive

$$J(u_t) = \mathbb{E}[(\beta u_t + d_t - T)^2].$$

Our objective is to minimize the function $J(u_t)$. Recall that after the calibration process, the posterior distribution of β is $\beta \sim N(\hat{\beta}, \sigma_{\tilde{\beta}}^2)$. Let $\tilde{\beta} = \beta - \hat{\beta}$, we have

$$J(u_t) = \mathbb{E}\left[\left(\hat{\beta} + \tilde{\beta}\right)u_t + d_t - T\right]^2.$$

Taking the derivative of $J(u_t)$ to $u_t$and set it to 0, we have

$$\frac{dJ(u_t)}{du} = 2\,\mathbb{E}\left[\left(\hat{\beta} + \tilde{\beta}\right)^2\right]u_t + 2\,\mathbb{E}\left[\left(\hat{\beta} + \tilde{\beta}\right)(d_t - T)\right]$$

$$= 2\,\mathbb{E}\left[\hat{\beta}^2 + 2\hat{\beta}\tilde{\beta} + \tilde{\beta}^2\right]u_t + 2\,\mathbb{E}\left[\left(\hat{\beta} + \tilde{\beta}\right)(d_t - T)\right] = 0.$$

Since $\mathbb{E}\left[\tilde{\beta}\right] = 0$,$\mathbb{E}\left[\tilde{\beta}^2\right] = \sigma_{\tilde{\beta}}^2$, u_t can be solved as

$$u_t = \frac{\hat{\beta}\left(T - \hat{d}_t\right)}{\hat{\beta}^2 + \sigma_{\tilde{\beta}}^2}.$$

A.2 Procedure for solving the optimization problem 15

The optimization problem in Equation 15 can be reformulated as the following standard bounded quadratic programming problem,

$$\min_{\mathbf{x}} \mathbf{x}^{\boldsymbol{T}}\mathbf{A}\mathbf{x} + 2\mathbf{b}^{\boldsymbol{T}}\mathbf{x} \tag{19}$$

subject to $\mathbf{m} \preccurlyeq \mathbf{x} \preccurlyeq \mathbf{M}$, where $\mathbf{x} = \mathbf{U}_{x,t}$, $\mathbf{b} = \widehat{\mathbf{D}}_x^T \mathbf{W}_F^T \mathbf{W}_F$, $\mathbf{A} = \mathbf{W}_F^T \mathbf{W}_F$, $\mathbf{m} = \underline{\mathbf{L}}(\mathbf{U}_x)$ and $\mathbf{M} = \overline{\mathbf{L}}(\mathbf{U}_x)$. The operator $\preccurlyeq$ represents that the left-hand side vector is element-wisely smaller than the right-hand side vector.

ADMM method can be used to solve this optimization problem with acceptable computational cost. The procedure is given in the following Algorithm.

Algorithm: ADMM for solving the bounded quadratic programming problem

---

Initiate $\mathbf{z} = \mathbf{u} = 0$. Select $\xi > 0$.

Do:

1. $\mathbf{x} \leftarrow (\mathbf{I} + \xi \mathbf{A})^{-1}(\mathbf{z} - \mathbf{u} - \xi \mathbf{b})$

2. $\mathbf{z} \leftarrow \Pi_C(\mathbf{x} + \mathbf{u}) = \min\{\max\{\mathbf{x}, \underline{\mathbf{L}}\}, \overline{\mathbf{L}}\}$, where C is the rectangle $\underline{\mathbf{L}} \preccurlyeq \mathbf{x} \preccurlyeq \overline{\mathbf{L}}$, and Π is the projection $\Pi_A(x) = \arg\min_{\mathbf{z} \in A} |\mathbf{x} - \mathbf{z}|$

3. $\mathbf{u} \leftarrow \mathbf{u} + \mathbf{x} - \mathbf{z}$

Until converge

Here, the parameter $\xi > 0$ controls the step size. After this parameter is selected, we shall calculate the Cholesky factorization of the matrix $\mathbf{I} + \xi \mathbf{A}$ beforehand and obtain the upper triangular matrix $\mathbf{T}$ such that $\mathbf{T}^T \mathbf{T} = \mathbf{I} + \xi \mathbf{A}$. In such a way, Step 1 is achieved by solving two linear systems $\mathbf{T}^T \mathbf{y} = \mathbf{z} - \mathbf{u} - \xi \mathbf{b}$ and then $\mathbf{T}\mathbf{x} = \mathbf{y}$.

The above ADMM algorithm is quite efficient in finding the solution. In this algorithm, the major computational burden in every iteration comes from solving two triangular linear

systems with 10 variables. We suggest running the algorithm for a fixed number, such as 10 steps, to restrict the control time. Hence, if we apply our bound algorithm to the APC system, this modification increases a very little amount of computational cost.

A3. Derivation of the regularized control law

The derivation of control law:

$$\mathbf{F}_x^{\boldsymbol{T}}\mathbf{F}_x = E\left[\mathbf{W}_F\mathbf{K}_{x,t}\right]^{\boldsymbol{T}}\left[\mathbf{W}_F\mathbf{K}_{x,t}\right]$$

$$= \mathbb{E}\left\{\left(\mathbf{U}_{x,t}^{\boldsymbol{T}}\left(\mathbf{I}+\widetilde{\mathbf{B}}_X\right)+\widehat{\mathbf{D}}_x^{\boldsymbol{T}}\right)\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\left(\left(\mathbf{I}+\widetilde{\mathbf{B}}_X\right)\mathbf{U}_{x,t}+\widehat{\mathbf{D}}_x\right)\right\}$$

$$= \mathbb{E}\left\{\mathbf{U}_{x,t}^{\boldsymbol{T}}\left(\mathbf{I}+\widetilde{\mathbf{B}}_X\right)\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\left(\mathbf{I}+\widetilde{\mathbf{B}}_X\right)\mathbf{U}_{x,t}+2\widehat{\mathbf{D}}_x^{\boldsymbol{T}}\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\left(\mathbf{I}+\widetilde{\mathbf{B}}_X\right)\mathbf{U}_{x,t}\right\}$$

$$= \mathbf{U}_{x,t}^{\boldsymbol{T}}\left(\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F+\mathbb{E}\left(\widetilde{\mathbf{B}}_X\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\widetilde{\mathbf{B}}_X\right)\right)\mathbf{U}_{x,t}+2\widehat{\mathbf{D}}_x^{\boldsymbol{T}}\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\mathbf{U}_{x,t}$$

Here, we used the fact that $\mathbb{E}\left[\widetilde{\mathbf{B}}_X\right]=0$. Setting the matrix $\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F+\mathbb{E}\left(\widetilde{\mathbf{B}}_X\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\widetilde{\mathbf{B}}_X\right)$ as $\mathbf{A}$ and the vector $\mathbf{W}_F^{\boldsymbol{T}}\mathbf{W}_F\widehat{\mathbf{D}}_x$ as $\mathbf{b}$, the optimization problem for the cautious regularized controller is transformed to the following bounded quadratic optimization problem.

$$\min_U \mathbf{U}_{x,t}^{\boldsymbol{T}}\mathbf{A}\mathbf{U}_{x,t}+2\mathbf{b}^{\boldsymbol{T}}\mathbf{U}_{x,t}$$

$$\text{subject to } \underline{\mathbf{L}}(\mathbf{U}_x)\le\mathbf{U}_{x,t}\le\overline{\mathbf{L}}(\mathbf{U}_x)$$

APPENDIX B. Supplementary Material for Chapter 3

B.1 Proof of Proposition 1

Proof. The proof of this borrows the idea from (Yan et al., 2019). The likelihood function can be minimized by,

$$vec(\mathcal{C}_j) = argmin_{\mathcal{C}_j} \left(R_j - (Z_j \otimes V_{jd} \dots \otimes V_{j1}) vec(\mathcal{C}_j)\right)^T (\Sigma_{d+1} \otimes \Sigma_d \dots \otimes \Sigma_1)^{-1} \Big(R_j - (Z_j \otimes V_{jd} \dots \otimes V_{j1}) vec(\mathcal{C}_j)\Big)$$

$$= argmin_{vec(\mathcal{C}_j)} vec(\mathcal{C}_j)^T \left(Z_j^T \Sigma_{d+1}^{-1} Z_j \otimes V_{jd}^T \Sigma_d^{-1} V_{jd} \otimes \dots \otimes V_{j1}^T \Sigma_1^{-1} V_{j1}\right) vec(\mathcal{C}_j) - 2vec(\mathcal{C}_j)^T (Z_j^T \Sigma_{d+1}^{-1} \otimes V_{jd}^T \Sigma_d^{-1} \dots \otimes V_{j1}^T \Sigma_1^{-1}) R_j$$

$$= \left(Z_j^T \Sigma_{d+1}^{-1} Z_j \otimes V_{jd}^T \Sigma_d^{-1} V_{jd} \otimes \dots \otimes V_{j1}^T \Sigma_1^{-1} V_{j1}\right)^{-1} (Z_j^T \Sigma_{d+1}^{-1} \otimes V_{jd}^T \Sigma_d^{-1} \dots \otimes V_{j1}^T \Sigma_1^{-1}) R_j$$

$$= \left(Z_j^T \Sigma_{d+1}^{-1} Z_j\right)^{-1} Z_j^T \Sigma_{d+1}^{-1} \otimes \left(V_{jd}^T \Sigma_d^{-1} V_{jd}\right)^{-1} V_{jd}^T \Sigma_d^{-1} \otimes \dots \otimes \left(V_{j1}^T \Sigma_1^{-1} V_{j1}\right)^{-1} V_{j1}^T \Sigma_1^{-1} R_j$$

Equivalently, this can be written in the tensor format as

$$\mathcal{C}_j = R_j \times_1 \left(V_{j1}^T \Sigma_1^{-1} V_{j1}\right)^{-1} V_{j1}^T \Sigma_1^{-1} \times_2 \left(V_{j2}^T \Sigma_2^{-1} V_{j2}\right)^{-1} V_{j2}^T \Sigma_2^{-1} \times_3 \dots \times_{d+1} \left(Z_j^T \Sigma_{d+1}^{-1} Z_j\right)^{-1} Z_j^T \Sigma_{d+1}^{-1}.$$

The (9) can be obtained from a similar procedure.

B.2 Proof of Proposition 2

Plugging in the estimation of $\mathcal{C}_j$, using the BCD algorithm, we have

$$\left(R_j - \left(Z_j \otimes V_{jd} \dots \otimes V_{j1}\right) vec\left(\mathcal{C}_j\right)\right)^T \left(\Sigma_{d+1} \otimes \Sigma_d \dots \otimes \Sigma_1\right)^{-1} \left(R_j - \left(Z_j \otimes V_{jd} \dots \otimes V_{j1}\right) vec\left(\mathcal{C}_j\right)\right)$$

$$= vec\left(\mathcal{C}_j\right)^T \left(Z_j^T \Sigma_{d+1}^{-1} Z_j \otimes V_{jd}^T \Sigma_d^{-1} V_{jd} \otimes \dots \otimes V_{j1}^T \Sigma_1^{-1} V_{j1}\right) vec\left(\mathcal{C}_j\right) -$$

$$2vec\left(\mathcal{C}_j\right)^T \left(Z_j^T \Sigma_{d+1}^{-1} \otimes V_{jd}^T \Sigma_d^{-1} \dots \otimes V_{j1}^T \Sigma_1^{-1}\right) R_j$$

$$= vec\left(\mathcal{C}_j\right)^T \left(Z_j^T \Sigma_{d+1}^{-1} Z_j \otimes I_d \dots \otimes I_1\right) vec\left(\mathcal{C}_j\right) -$$

$$2vec\left(\mathcal{C}_j\right)^T \left(Z_j^T \Sigma_{d+1}^{-1} \otimes V_{jd}^T \Sigma_d^{-1} \dots \otimes V_{j1}^T \Sigma_1^{-1}\right) R_j$$

$$= -vec\left(\mathcal{C}_j\right)^T \left(\Sigma_{d+1}^{-1} Z_j \left(Z_j^T \Sigma_{d+1}^{-1} Z_j\right)^{-1} Z_j^T \Sigma_{d+1}^{-1}\right) \otimes \left(\Sigma_d^{-1} V_{jd} V_{jd}^T \Sigma_d^{-1}\right) \otimes \dots \otimes \left(\Sigma_1^{-1} V_{j1} V_{j1}^T \Sigma_1^{-1}\right) R_j$$

$$= -\left\| X_{j,d+1} \otimes \left(V_{jd}^T \Sigma_d^{-1}\right) \otimes \dots \otimes \left(V_{j1}^T \Sigma_1^{-1}\right) R_j \right\|^2$$

$$= -\left\| R_j \times_1 V_{j1}^T \Sigma_1^{-1} \times_2 V_{j2}^T \Sigma_2^{-1} \times_3 \dots \times_d V_{jd}^T \Sigma_d^{-1} \times_{d+1} X_{j,d+1} \right\|^2$$

For the proof of (12), the procedure will be exactly the same as the above procedure and we just skip it. Then, we will discuss how to maximize (11) and (12). W.L.O.G, we will have

$$\left\| R_j \times_1 V_{j1}^T \Sigma_1^{-1} \times_2 V_{j2}^T \Sigma_2^{-1} \times_3 \dots \times_d V_{jd}^T \Sigma_d^{-1} \times_{d+1} X_{j,d+1} \right\|^2 = \left\| \mathcal{W}_{jk} \times_k \tilde{V}_{jk} \Sigma_k^{-1/2} \right\|^2$$

$$= \left\| \tilde{V}_{jk} \Sigma_k^{-1/2} \boldsymbol{W}_{jk} \right\|^2$$

It is not hard to prove that $\max_{\tilde{V}_{jk}} \left\| \tilde{V}_{jk} \Sigma_k^{-1/2} \boldsymbol{W}_{jk} \right\|^2$ s.t. $\tilde{V}_{jk}^T \tilde{V}_{jk} = I$ can be solved by the first P_k eigenvectors of $\Sigma_k^{-1/2} \boldsymbol{W}_{jk}$.

B.3 Proof of Proposition 3

Proof. Minimizing (15) can be transformed into

$$J(\mathcal{X}_t) = \min_{\mathcal{X}_t} E\|\mathcal{Y}_{t+1}(\mathcal{X}_t) - T\|_F^2$$

$$= \min_{\mathcal{X}_t} E\left\|\mathcal{Y}_{t+1}(\mathcal{X}_t) - E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right) + E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right) - T\right\|_F^2$$

$$= \min_{\mathcal{X}_t}\left\|\mathcal{Y}_{t+1}(\mathcal{X}_t) - E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right)\right\|_F^2 + \left\|E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right) - T\right\|_F^2$$
$$+ 2E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t) - E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right)\right)\left(E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right) - T\right)$$

$$= \min_{\mathcal{X}_t}\left\|\mathcal{Y}_{t+1}(\mathcal{X}_t) - E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right)\right\|_F^2 + \left\|E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right) - T\right\|_F^2$$

Since $\left\|\mathcal{Y}_{t+1}(\mathcal{X}_t) - E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right)\right\|_F^2$ is the variance of $\mathcal{Y}_{t+1}(\mathcal{X}_t)$ and cannot be minimized by the control variable, then we will have

$$E\left(\mathcal{Y}_{t+1}(\mathcal{X}_t)\right) = T.$$

This closes the first half part of the proof. Then, we will discuss how to recover $\mathcal{X}_t$ from (15).From the definition we will have:

$$Bvec(\mathcal{X}_t) = vec(R_{Bt}),$$

Where $B \in R^{Q\times P}$ is an unfolding of tensor B with $P = \Pi_{i=1}^{l} P_i$ and $Q = \Pi_{i=1}^{d} Q_d$. The SVD of B is $B = (V_{Bd}\otimes \ldots V_{B2}\otimes V_{B1})C_B(U_{Bl}\otimes \ldots \otimes U_{B1})^T$. Therefore, we have

$$vec(\mathcal{X}_t) = (U_{Bl}\otimes \ldots \otimes U_{B1})C_B^{-1}(V_{Bd}\otimes \ldots V_{B2}\otimes V_{B1})^T vec(R_{Bt}),$$

Recall that we set $U_{B1}, \dots, U_{Bl}$ equal to the identity matrix. Then we have

$$vec(\mathcal{X}_t) = C_B^{-1}(V_{Bd} \otimes \dots V_{B2} \otimes V_{B1})^T vec(R_{Bt}).$$

B.4 Proof of proposition 4

Proof. Recall that the process model is

$$\mathcal{Y}_t = \Sigma_{j=1}^{p} \mathcal{Y}_{t-j} * \mathcal{A}_j + \Sigma_{n=0}^{l-1} \mathcal{X}_{t-n} * \mathcal{B}_n + \delta E_t,$$

Define $\xi_t = \begin{bmatrix} \mathcal{Y}_t \\ \vdots \\ \mathcal{Y}_{t-p+1} \end{bmatrix} \in R^{pQ_1 \times Q_2 \times \dots \times Q_d}, \widetilde{\mathcal{X}}_t = \begin{bmatrix} \mathcal{X}_t \\ \vdots \\ \mathcal{X}_{t-p+1} \end{bmatrix} \in R^{pP_1 \times P_2 \times \dots \times P_s}$. Then if we define $\mathcal{A}^* = \begin{bmatrix} \mathcal{A}_1 \dots \mathcal{A}_{p-1} & \mathcal{A}_p \\ \boldsymbol{I} & \boldsymbol{0} \end{bmatrix} \in R^{pQ_1 \dots Q_d \times pQ_1 \dots Q_d}, \mathcal{B}^* = \begin{bmatrix} \mathcal{B}_0 \dots \mathcal{B}_{l-1} & 0 \\ \boldsymbol{I} & \boldsymbol{0} \end{bmatrix} \in R^{pQ_1 \dots Q_d \times pP_1 \dots P_s}$. Then we will have

$$\xi_t = \mathcal{A}^* * \xi_{t-1} + \mathcal{B}^* * \widetilde{\mathcal{X}}_t + \delta \tilde{E}_t$$

$$\mathcal{Y}_t = C_{\mathcal{Y}_t} \times_1 U_1 \times_2 U_2 \times \dots \times_d U_d$$

$$\mathcal{A}_i = C_{\mathcal{A}_i} \times_1 V_{i1} \times_2 V_{i2} \dots \times_d V_{id} \times_{d+1} U_1 \times_{d+2} U_2 \times \dots \times_{2d} U_d$$

$$\mathcal{B}_i = C_{\mathcal{B}_i} \times_1 V_{B_i 1} \times_2 V_{B_i 2} \dots \times_d V_{Bd} \times_{d+1} U_{B_i 1} \times_{d+2} U_{B_i 2} \times \dots \times_{d+s} U_{B_i s}$$

$C_{\mathcal{Y}_t} \in R^{q_1 \times q_2 \times \dots \times q_d \times q_1 \times q_2 \times \dots \times q_d}, U_1 \in R^{Q_1 \times q_1}, U_2 \in R^{Q_2 \times q_2} \dots$, where $q_1, q_2 \ll Q_1, Q_2$. Then, we have

$$\xi_t = \mathcal{A}^0 * \xi_{t-1} + \mathcal{B}^0 * \widetilde{\mathcal{X}}_t + \delta \tilde{E}_t$$

$$\xi_t = \begin{bmatrix} \tilde{C}_{y_t} \\ \vdots \\ \tilde{C}_{y_{t-p+1}} \end{bmatrix} \times_1 U_1 \times_2 U_2 \times \ldots \times_d U_d,$$

$$\mathcal{A}^0 = \begin{bmatrix} \tilde{C}_{\mathcal{A}_1} \times_1 V_{11} \ldots \times_d V_{1d} \times_{d+1} U_1 \ldots \times_{2d} U_d \ldots & \tilde{C}_{\mathcal{A}_p} \times_1 V_{p1} \ldots \times_d V_{pd} \times_{d+1} U_1 \times_{d+2} \ldots \times_{2d} U_d \\ \boldsymbol{I} & \mathbf{0} \end{bmatrix}$$

$$\mathcal{B}^0 = \begin{bmatrix} \tilde{C}_{\mathcal{B}_0} \times_1 V_{B_0 1} \ldots \times_d V_{B_0 d} \times_{d+1} U_{B_0 1} \ldots \times_{d+s} U_{B_0 s} \ldots \tilde{C}_{\mathcal{B}_{l-1}} \times_1 V_{B_{l-1} 1} \ldots \times_d V_{B_{l-1} d} \times_{d+1} U_{B_{l-1} 1} \ldots \times_{d+s} U_{B_{l-1} s} & 0 \\ \boldsymbol{I} & \mathbf{0} \end{bmatrix}$$

We have

$$vec(\mathcal{Y}_t) = (U_d \otimes \ldots \otimes U_1) vec(\tilde{C}_{y_t})$$

And

$$vec(\xi_t) = (U_d \otimes \ldots \otimes U_1) \begin{bmatrix} vec(\tilde{C}_{y_t}) \\ \vdots \\ vec(\tilde{C}_{y_{t-p+1}}) \end{bmatrix}$$

Since U_i is known and $U_i^T U_i = I$. We have

$$(U_d \otimes \ldots \otimes U_1)^T vec(\xi_t) = (U_d \otimes \ldots \otimes U_1)^T (U_d \otimes \ldots \otimes U_1) \begin{bmatrix} vec(C_{y_t}) \\ \vdots \\ vec(C_{y_{t-p+1}}) \end{bmatrix}$$

$$= \begin{bmatrix} vec(C_{y_t}) \\ \vdots \\ vec(C_{y_{t-p+1}}) \end{bmatrix}$$

Vectorize both side of (1) and time $(U_2 \otimes U_1)^T$ on both side we have

$$(U_d \otimes \dots \otimes U_1)^T vec(\xi_t)$$

$$= (U_d \otimes \dots \otimes U_1)^T vec(\mathcal{A}^0 * \xi_{t-1}) + (U_d \otimes \dots \otimes U_1)^T vec\left(\mathcal{B}^0 * \widetilde{\mathcal{X}}_t\right)$$

$$+ (U_d \otimes \dots \otimes U_1)^T vec(\delta \tilde{E}_t)$$

This is equivalent to

$$\begin{bmatrix} vec\left(\tilde{C}_{y_t}\right) \\ \vdots \\ vec\left(\tilde{C}_{y_t-p+1}\right) \end{bmatrix} =$$

$$(U_d \otimes \dots U_1)^T \begin{bmatrix} (V_{1d} \otimes \dots V_{11})\tilde{C}_{\mathcal{A}_1}(U_d \otimes \dots U_1)^T & \dots & (V_{pd} \otimes \dots V_{p1})\tilde{C}_{\mathcal{A}_p}(U_d \otimes \dots U_1)^T & 0 \\ \boldsymbol{I} & & & \boldsymbol{0} \end{bmatrix}$$

$$(U_d \otimes \dots U_1)\begin{bmatrix} vec\left(C_{y_{t-1}}\right) \\ \vdots \\ vec\left(C_{y_t-p}\right) \end{bmatrix} +$$

$$(U_d \otimes \dots U_1)^T \begin{bmatrix} (V_{B_0 d} \otimes \dots V_{B_0 1})\tilde{C}_{B_0}\left(U_{B_0 s} \otimes \dots U_{B_0 1}\right)^T & \dots & (V_{B_{l-1} d} \otimes \dots V_{B_{l-1} 1})\tilde{C}_{B_{l-1}}\left(U_{B_{l-1} s} \otimes \dots U_{B_{l-1} 1}\right)^T & 0 \\ \boldsymbol{0} & & & \boldsymbol{0} \end{bmatrix}$$

$$\begin{bmatrix} (U_{B_0 s} \otimes \dots U_{B_0 1})vec\left(C_{\mathcal{X}_t}\right) \\ \vdots \\ (U_{B_{l-1} s} \otimes \dots U_{B_{l-1} 1})vec\left(C_{\mathcal{X}_{t-1+1}}\right) \\ vec(\mathcal{X}_{t-l}) \\ \vdots \\ vec\left(\mathcal{X}_{t-p+1}\right) \end{bmatrix} + (U_d \otimes \dots \otimes U_1)^T vec(\delta \tilde{E}_t)$$

This is equivalent to

$$vec(\xi_t) = \tilde{A} vec(\xi_{t-1}) + \tilde{B}\widetilde{\mathcal{X}}_t + (U_d \otimes \dots \otimes U_1)^T vec\left(\delta \tilde{E}_t\right)$$

Take expectation, we have

$$E\{vec(\xi_t)\} = \tilde{A} E\{vec(\xi_{t-1})\} + \tilde{B} E\{\widetilde{\mathcal{X}}_t\}$$

This is a typical state-space model. The controllability condition of this state-space model is $\left[\tilde{B}\ \tilde{A}\tilde{B}\ \tilde{A}^2\tilde{B}\ \ldots \tilde{A}^{pq_1\cdots q_d-1}\tilde{B}\right]$ has full-row rank.

B.5 Data generation scheme

In semiconductor manufacturing, there are two different coordinate systems: the wafer-level coordinate system and the field-level coordinate system as shown in Fig.2, we define a wafer-level coordinate system (X, Y) whose origin is at the center of the wafer and the Y-axis is perpendicular to the flat edge of a wafer. Within each field, we also define a field-level coordinate (x, y) with the origin at the center of this field, and the x-axis of the field-level coordinate is aligned with the X-axis in the wafer-level coordinate system. From the manufacturing process, the overlay error can be decomposed into three major sources:

- *The wafer-level errors*. This type of error affects the quality of the entire wafer and usually originated from the stage control error and wafer distortion. The relationship between this kind of error and wafer-level coordinate is injective. It can be represented by a function mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$, $\left(F_x(X,Y), F_y(X,Y)\right)^T$, where $F_x(X,Y), F_y(X,Y)$ represent the wafer-level error along the X and Y axes at this location.
- *The common field-level errors.* The common field-level errors affect all the fields on a wafer, which is associated with the exposure system. Since all of the fields on a wafer are fabricated by the same exposure system, the common field-level error share the same overlay error pattern within each field and thus can be represented as $\left(f_x(x,y), f_y(x,y)\right)^T$.
- *The individual field-level errors.* Some chance causes will result in the different error patterns among different fields, which is known as individual field errors. For field i,

the individual field level error is represented by $\left(f_{i,x}(x,y), f_{i,y}(x,y)\right)^T$.

The summation of these three types of errors is the total overlay error denoted as (Δ_x, Δ_y), which can be represented by:

$$\Delta_x(X,Y,x,y,i) = F_x(X,Y) + f_x(x,y) + f_{i,x}(x,y), \tag{15a}$$

$$\Delta_y(X,Y,x,y,i) = F_y(X,Y) + f_y(x,y) + f_{i,y}(x,y). \tag{15b}$$

In practice, the wafer-level overlay error and common-field level overlay error can be further decomposed by third-order polynomial basis as shown in Equations 19 and 20 (Huang et al. 2008).The control variables $U_1, \ldots U_{20}$,$u_1, \ldots u_{20}$ is designed to control the corresponding overlay coefficients $K_1, \ldots, K_{20}$ and $k_1, \ldots, k_{20}$ individually. However, in the real application, since the lithography machine is a very complex system, the change of each control variable will have some unknown side effects. Therefore, it is better for us to build one model between all control variables and total overlay error measurements rather than build multiple models between each control variable and the corresponding overlay coefficient.

$$\begin{cases} F_x(X,Y) = K_1 + K_3X + K_5Y + K_7X^2 + K_9XY + K_{11}Y^2 \\ \qquad\qquad +K_{13}X^3 + K_{15}X^2Y + K_{17}XY^2 + K_{19}Y^3 \\ F_y(X,Y) = K_2 + K_4X + K_6Y + K_8X^2 + K_{10}XY + K_{12}Y^2 \\ \qquad\qquad +K_{14}X^3 + K_{16}X^2Y + K_{18}X\,Y^2 + K_{20}Y^3 \end{cases}; \tag{19}$$

$$\begin{cases} f_x(x,y) = k_1 + k_3x + k_5y + k_7x^2 + k_9xy + k_{11}y^2 \\ \qquad\qquad +k_{13}x^3 + k_{15}x^2y + k_{17}xy^2 + k_{19}y^3 \\ f_y(x,y) = k_2 + k_4x + k_6y + k_8x^2 + k_{10}xy + k_{12}y^2 \\ \qquad\qquad +k_{14}x^3 + k_{16}x^2y + k_{18}xy^2 + k_{20}y^3 \end{cases}; \tag{20}$$

The overlay measurements can be presented as $N * 2$ image data, where N is the total number of overlay measurements on each wafer. In this case study, overlay data are generated from a simulator endorsed by a well-known semiconductor company. Due to the confidential issue, we cannot elaborate on this simulator in great detail. But in general, we first generate overlay coefficients $K_1, \dots, K_{20}$, $k_1, \dots, k_{20}$ from a multivariate time series model i.e., the ARIMA(1,1,1) model. Then, we can recover the wafer-level and common field-level overlay error by using the third-order polynomial basis. By aggregating these two types of errors, the final overlay measurement can be obtained. In our simulator, we set all individual field level errors to be zero, because the magnitude of them is comparatively much smaller than wafer-level error and common field-level error.

REFERENCES

Anandkumar, A., Ge, R., Hsu, D., Kakade, S.M. and Telgarsky, M., (2015). Tensor decompositions for learning latent variable models (A survey for ALT). In *International Conference on Algorithmic Learning Theory*. Springer, Cham 19-38.

Apley, D.W. and Kim, J., (2004). Cautious control of industrial process variability with uncertain input and disturbance model parameters. *Technometrics*, 46(2), 188-199.

Armitage Jr, J.D., and Kirk, J.P., (1988), January. Analysis of overlay distortion patterns. In *Integrated Circuit Metrology, Inspection, and Process Control II*. International Society for Optics and Photonics. 921, 207-223

Bar-Shalom, Y., (1981). Stochastic dynamic programming: Caution and probing. *IEEE Transactions on Automatic Control*, 26(5), 1184-1195.

Bollen, B., (2015). What should the value of lambda be in the exponentially weighted moving average volatility model?. *Applied Economics*, 47(8), 853-860.

Boyd, S., Parikh, N. and Chu, E., (2011). *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc.

Brunner, T.A., Menon, V.C., Wong, C.W., Gluschenkov, O., Belyansky, M.P., Felix, N.M., Ausschnitt, C.P., Vukkadala, P., Veeraraghavan, S. and Sinha, J.K., (2013). Characterization of wafer geometry and overlay error on silicon wafers with nonuniform stress. *Journal of Micro/Nanolithography, MEMS, and MOEMS*, 12(4), 043002.

Burdisso, R. A., & Haftka, R. T. (1989). Optimal location of actuators for correcting distortions in large truss structures. *AIAA Journal*, *27*(10), 1406-1411.

Burdisso, R. A., & Haftka, R. T. (1990). Statistical analysis of static shape control in space structures. *AIAA Journal*, *28*(8), 1504-1508.

Chang, C.C., Pan, T.H., Wong, D.S.H. and Jang, S.S., (2012). An adaptive-tuning scheme for G&P EWMA run-to-run control. *IEEE transactions on semiconductor manufacturing*, 25(2), 230-237.

Chee, C. Y., Tong, L., & Steven, G. P. (2002). Static shape control of composite plates using a slope-displacement-based algorithm. *AIAA Journal*, *40*(8), 1611-1618.

Chen, G., McAvoy, T.J. and Piovoso, M.J., (1998). A multivariate statistical controller for on-line quality improvement. *Journal of Process Control*, 8(2), 139-149.

Chien, C.F., Chen, Y.J., Hsu, C.Y. and Wang, H.K., (2013). Overlay error compensation using advanced process control with dynamically adjusted proportional-integral R2R controller. *IEEE Transactions on Automation Science and Engineering*, 11(2), pp.473-484.

Del Castillo, E. and Hurwitz, A.M., (1997). Run-to-run process control: Literature review and extensions. *Journal of Quality Technology*, 29(2), 184-196.

Del Castillo, E. and Rajagopal, R., (2002). A multivariate double EWMA process adjustment scheme for drifting processes. *Iie Transactions*, 34(12), 1055-1068.

Du, J., Liu, C., Liu, J., Zhang, Y., & Shi, J. (2021). Optimal Design of Fixture Layout for Compliant Part With Application in Ship Curved Panel Assembly. *Journal of Manufacturing Science and Engineering*, *143*(6)

Du, J., Yue, X., Hunt, J. H., & Shi, J. (2019). Optimal placement of actuators via sparse learning for composite fuselage shape control. *Journal of Manufacturing Science and Engineering*, *141*(10).

Gahrooei, M.R., Yan, H., Paynabar, K. and Shi, J., (2018). A novel approach for fusion of heterogeneous sources of data. *arXiv preprint arXiv:1803.00138*.

Gates, D. (2007). Boeing finds 787 pieces aren't quite a perfect fit. *Rapport technique, Seattle Times*.

Grant, M., Boyd, S., & Ye, Y. (2008). CVX: Matlab software for disciplined convex programming.

Good, R. and Qin, S.J., (2002), May. Stability analysis of double EWMA run-to-run control with metrology delay. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301)* IEEE. 3, 2156-2161.

Haftka, R. T., & Adelman, H. M. (1985). An analytical investigation of shape control of large space structures by applied temperatures. *AIAA Journal*, *23*(3), 450-457.

Haftka, R. T., & Adelman, H. M. (1985). Selection of actuator locations for static shape control of large space structures by heuristic integer programing. In *Advances and Trends in Structures and Dynamics* (pp. 575-582). Pergamon.

Hakim, S., & Fuchs, M. B. (1996). Quasistatic optimal actuator placement with minimum worst-case distortion criterion. *AIAA Journal*, *34*(7), 1505-1511.

Hannan, E.J., Dunsmuir, W.T. and Deistler, M., (1980). Estimation of vector ARMAX models. *Journal of Multivariate Analysis*, 10(3), 275-295.

Hitchcock, F.L., (1927). The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1-4), 164-189.

Huang C.-C. K. and Tien D. (2008), Overlay control goes to high-order, *Microlithography World.*

Huang, C.Y., Chiu, C.F., Wu, W.B., Shih, C.L., Huang, C.C.K., Huang, H., Choi, D., Pierson, B. and Robinson, J.C., (2012). Overlay control methodology comparison: field-by-field and high-order methods. In *Metrology, Inspection, and Process Control for Microlithography XXVI* International Society for Optics and Photonics 8324, 832427.

Kiers, H.A., (2000). Towards a standardized notation and terminology in multiway analysis. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 14(3), 105-122.

Kohnke, P. (2013). ANSYS mechanical APDL theory reference. *Canonsburg, PA, USA: ANSYS Inc*.

Kolda, T.G. and Bader, B.W., (2009). Tensor decompositions and applications. *SIAM review*, 51(3), 455-500.

Liu, C., Law, A.C.C., Roberson, D. and Kong, Z.J., (2019). Image analysis-based closed-loop quality control for additive manufacturing with fused filament fabrication. *Journal of Manufacturing Systems*, 51, 75-86.

Li, X., Xu, D., Zhou, H. and Li, L., (2018). Tucker tensor regression and neuroimaging analysis. *Statistics in Biosciences*, 10(3), 520-545.

Lock, E.F., (2018). Tensor-on-tensor regression. *Journal of Computational and Graphical Statistics*, 27(3), 638-647.

Maciejowski, J. M. (2002). *Predictive control: with constraints*. Pearson education.

Natarajan, B. K. (1995). Sparse approximate solutions to linear systems. *SIAM journal on computing*, *24*(2), 227-234.

Park, S.J., Lee, M.S., Shin, S.Y., Cho, K.H., Lim, J.T., Cho, B.S., Jei, Y.H., Kim, M.K. and Park, C.H., (2005). Run-to-run overlay control of steppers in semiconductor manufacturing systems based on historical data analysis and neural network modeling. *IEEE transactions on semiconductor manufacturing*, 18(4), 605-613.

Plumbridge, W. J., Matela, R. J., & Westwater, A. (2007). *Structural integrity and reliability in electronics: enhancing performance in a lead-free environment*. Springer Science & Business Media.

Ponslet, E., HAFTKA, R., & CUDNEY, H. (1993, January). Optimal placement of tuning masses on truss structures by genetic algorithms. In *34th Structures, Structural Dynamics and Materials Conference* (p. 1586).

Reddy, J. N. (2019). *Introduction to the finite element method*. McGraw-Hill Education.

Tseng, S.T., Chou, R.J. and Lee, S.P., (2002). A study on a multivariate EWMA controller. *Iie Transactions*, 34(6), 541-549.

Tseng, S.T., Mi, H.C. and Lee, I.C., (2016). A multivariate EWMA controller for linear dynamic processes. *Technometrics*, 58(1), 104-115.

Wen, Y., Yue, X., Hunt, J. H., & Shi, J. (2018). Feasibility analysis of composite fuselage shape control via finite element analysis. *Journal of manufacturing systems*, *46*, 272-281.

Yan, H., Paynabar, K. and Shi, J., (2014). Image-based process monitoring using low-rank tensor decomposition. *IEEE Transactions on Automation Science and Engineering*, 12(1), 216-227.

Yan, H., Paynabar, K., & Pacella, M. (2019). Structured point cloud data analysis via regularized tensor regression for process modeling and optimization. *Technometrics*, 61(3), 385-395.

Yue, X., Wen, Y., Hunt, J.H. and Shi, J., (2018). Surrogate model-based control considering uncertainties for composite fuselage assembly. *Journal of Manufacturing Science and Engineering*, 140(4).

Zhou, H., Li, L. and Zhu, H. (2013). Tensor Regression with Applications in Neuroimaging Data Analysis. *Journal of the American Statistical Association*, 108(502), 540-552.

Zhong, J., Liu, J., & Shi, J. (2010). Predictive control considering model uncertainty for variation reduction in multistage assembly processes. *IEEE Transactions on Automation Science and Engineering*, *7*(4), 724-735.